Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

LINEAR NOISE APPROXIMATION ASSISTED BAYESIAN INFERENCE ON MECHANISTIC MODEL OF PARTIALLY OBSERVED STOCHASTIC REACTION NETWORK

Wandi Xu[1], and Wei Xie[1]

[1]Dept. of Mechanical and Industrial Eng., Northeastern University, Boston, MA, USA

ABSTRACT

To support mechanism online learning and facilitate digital twin development for biomanufacturing processes, this paper develops an efficient Bayesian inference approach for partially observed enzymatic stochastic reaction network (SRN), a fundamental building block of multi-scale bioprocess mechanistic model. To tackle the critical challenges brought by the nonlinear stochastic differential equations (SDEs)-based mechanistic model with partially observed state and having measurement errors, an interpretable Bayesian updating linear noise approximation (LNA) metamodel, incorporating the structure information of the mechanistic model, is proposed to approximate the likelihood of observations. Then, an efficient posterior sampling approach is developed by utilizing the gradients of the derived likelihood to speed up the convergence of MCMC. The empirical study demonstrates that the proposed approach has a promising performance.

1 INTRODUCTION

Partially observed stochastic reaction network (SRN) modeling the dynamics of a population of interacting species, such as chemical molecules participating in multiple reactions, is the fundamental building block of *multi-scale bioprocess mechanistic model* characterizing the causal interdependences from molecular- to macro-kinetics. It plays a critical role to: (1) facilitate digital twin development and support mechanism learning for biomanufacturing processes; (2) allow us to probe critical latent state based on partially observed information; and (3) serve as a fundamental model for a biofoundry platform (Hillson et al. 2019) that can integrate heterogeneous online and offline measures collected from different manufacturing processes and speed up the bioprocess development with much less experiments. Model inference on the SRN mechanistic model based on heterogeneous data also helps to strengthen the theoretical foundations of federated learning on bioprocess mechanisms, through which we can train and advance knowledge.

SRN has three key features that make the model inference challenging. First, the continuous-time state transition model, representing the evolution of concentration or number of molecules, is highly nonlinear. At any time, the reaction rates, characterizing the regulation mechanisms of enzymatic reaction network, are a function of random state. We adopt the diffusion approximation in Gillespie (2000) to model the state dynamics with a set of coupled stochastic differential equations (SDEs). In this case, the state transition model has double-stochasticity, making it analytically intractable to obtain the state transition densities at different times and also hard to get the closed form likelihood of observations. Second, since the state is partially observed, we need to integrate out the unobserved state variables to get the likelihood. Third, the data collected from biomanufacturing processes are heterogeneous and also subject to measurement errors.

The model inference of enzymatic SRN has found increasing interest especially in biomanufacturing digital twin development. Even under the situations with the reaction network structure known, that is built on thousand years of the understanding on biological system mechanisms, the mechanistic model parameters are often unknown. It is necessary to infer these parameters using the observations collected from biomanufacturing processes. Since each batch of production can be expensive, we often have very small amount of experimental observations. Coupled with high inherent stochasticity of biomanufacturing processes, the model uncertainty tends to be high. However, frequentist model uncertainty quantification

approaches are built on asymptotic approximation, such as asymptotic normality and bootstrap. Thus, in this paper, we focus on a Bayesian inference on multi-scale mechanistic model, which can support online learning and interpretability.

An enormous volume of literature has been dedicated to Bayesian inference for SRN mechanistic model. As the exact state transition density of the SDEs-based mechanistic model is unknown, coupled with another challenge (i.e., the partially observed state), the marginal likelihood integrating out the unobserved state variables is intractable. Thus, many existing works are sampling approaches without the explicit calculations of the likelihood, such as approximate Bayesian computation (ABC) and its variants (Xie et al. 2022). But in sampling approaches without using likelihood, the complex structure and high stochasticity of SRN make the simulation generating a large amount of sample paths computationally expensive and the acceptance rates of samplers very low. *Therefore, we construct a metamodel to approximate the state transition densities of the SDEs, obtain a likelihood approximation, and utilize it to speed up Bayesian inference.*

Gaussian Process (GP) is often used as a metamodel. Archambeau et al. (2007) and Garcia et al. (2017) use GPs as priors for nonparametric estimation of the drift and diffusion terms of SDEs without an exact knowledge of their functional forms. In this paper, we suppose the structure of SDEs-based mechanistic model is known. To completely exploit such structure information and improve the interpretability of constructed metamodel, we refer to the deterministic ordinary differential equation (ODE)-based dynamic system inference. In particular, Yang et al. (2021) specify a GP prior over the solution to the ODE, and restrict the GP on a manifold that satisfies the ODE system, to address the incompatibility between the metamodel and the mechanistic model. And an alternative to GP under SDE-based model is linear noise approximation (LNA). The LNA was originally proposed to approximate the solution of the chemical master equation (CME), and it can be derived in a number of ways. For instance, Ferm et al. (2008) and Ruttor and Opper (2009) follow the idea of an asymptotic system size expansion, and derive the LNA by approximating the CME through a Taylor expansion. Since the solution of SDE itself is a random variable, it is difficult to extend the GP approach developed in Yang et al. (2021) to the SDE model inference. Instead, following Fearnhead et al. (2014), we specify the derived LNA as a prior to the solution of the SDE, through which we take full advantage of the structure information provided by the SDE model without the time-consuming numerical integration.

The likelihood of observations can be obtained under the LNA, but the exact Bayesian posterior is still not analytically tractable as a conjugate prior is hard to find. One thus defers to sampling approaches to generate samples from the posterior. The most common one is Markov chain Monte Carlo (MCMC), such as Metropolis-Hastings algorithm. Its effectiveness depends heavily on the choice of the proposal distribution. Metropolis-adjusted Langevin algorithm (MALA) makes use of the additional gradient information of the target posterior distribution to construct a better proposal distribution, which is shown to have a faster mixing time compared with classic MCMC (Chewi et al. 2021). Therefore, in this paper, we specifically tailor a MALA procedure to generate posterior samples more efficiently.

In specific, we propose a LNA assisted Bayesian inference on the nonlinear multivariate SDE-based mechanistic model with partially observed state and subject to measurement errors. The main contributions are twofold. First, an interpretable Bayesian updating LNA metamodel is developed for likelihood approximation. It provides a coherent way to simultaneously satisfy the SDE model and fit the observed data, allowing us to probe critical latent state based on partially observed information. Second, the proposed MALA procedure utilizes the gradient information from the derived likelihood to speed up MCMC search and more efficiently generate posterior samples. The proposed Bayesian inference for SRN can support online mechanism learning, facilitate digital twin development, and speed up bioprocess design and control.

The paper is organized as follows. We provide a brief introduction of the SDE-based mechanistic model for enzymatic SRN and problem description in Section 2. To facilitate the model Bayesian inference, the LNA is used to construct the state transition densities and a closed form likelihood is thus derived in Section 3. Then, we propose an efficient and interpretable Bayesian posterior sampling algorithm in Section 4. Its performance is studied in Section 5. Finally, we conclude the paper in Section 6.

2 STOCHASTIC REACTION NETWORK (SRN) MODEL AND PROBLEM DESCRIPTION

(1) SRN model. We first review a general SRN composed of J species, denoted by $\boldsymbol{X} = (X_1, X_2, \ldots, X_J)^\top$, interacting with each other through K reactions. The number of molecules of species j at time t is denoted by $x_j(t)$ and $\boldsymbol{x}(t) = (x_1(t), x_2(t), \ldots, x_J(t))^\top$. Each reaction is characterized by a nonzero reaction vector $\boldsymbol{C}_k \in \mathbb{R}^J$ for $k = 1, 2 \ldots, K$, describing the change in the numbers of J species' molecules when a k-th molecular reaction occurs. The associated propensity function, denoted by ω_k, describes the probability with which the k-th reaction occurs per time unit. Specifically, for the k-th reaction equation given by

$$p_{k1}X_1 + p_{k2}X_2 + \cdots + p_{kJ}X_J \xrightarrow{\omega_k} q_{k1}X_1 + q_{k2}X_2 + \cdots + q_{kJ}X_J,$$

the reaction relational structure, specified by $\boldsymbol{C}_k = (q_{k1} - p_{k1}, q_{k2} - p_{k2}, \ldots, q_{kJ} - p_{kJ})^\top$, is known for $k = 1, 2, \ldots, K$. Thus, the *stoichiometry matrix* $\boldsymbol{C} = (\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_K) \in \mathbb{R}^{J \times K}$ characterizes the structure information of the reaction network composed of K reactions, where its (i, j)-th element represents the number of molecules of the i-th species that are either consumed (indicated by a negative value) or produced (indicated by a positive value) in each random occurrence of the j-th reaction.

Then, we describe the state transition model for bioprocess. As a multi-scale bioprocess representing the dependence from molecular- to macro-kinetics, it is built on the fundamental building block, i.e., molecular reaction network. Let $d\boldsymbol{R}(t)$ represent a K-dimensional vector of occurrences of each molecular reaction in an infinitesimal time interval $(t, t + dt]$. It follows a distribution with parameters depending on the propensity functions $\boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta}) = (\omega_1(\boldsymbol{x}(t); \boldsymbol{\theta}_1), \omega_2(\boldsymbol{x}(t); \boldsymbol{\theta}_2), \ldots, \omega_K(\boldsymbol{x}(t); \boldsymbol{\theta}_K))^\top$, where the structure of each $\omega_k(\boldsymbol{x}(t); \boldsymbol{\theta}_k)$, characterizing the bioprocess regulation mechanism for the k-th molecular reaction, is given and we focus on the inference of the unknown parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^\top, \boldsymbol{\theta}_2^\top, \ldots, \boldsymbol{\theta}_K^\top)^\top$.

Due to the fact that reaction events change species numbers by an integer amount, the state transition model is naturally characterized by a continuous-time Markov jump process (Anderson and Kurtz 2011). In particular, assuming that two reactions cannot occur at exactly the same time, one can represent the occurrences number of each k-th reaction in an infinitesimal time interval $(t, t + dt]$, denoted by $dR_k(t)$ (i.e., the k-th component of $d\boldsymbol{R}(t)$), using one of the most elementary counting process, namely, the nonhomogeneous Poisson process. Since the dynamic change of propensity function in any infinitesimal time interval $(t, t + dt]$ is negligible, the intensity of $dR_k(t)$ becomes $\omega_k(\boldsymbol{x}(t); \boldsymbol{\theta}_k)dt$. And conditional on $\boldsymbol{x}(t)$, $dR_k(t)$ for $k = 1, 2, \ldots, K$ can be considered as independent of one another and are Poisson$(\omega_k(\boldsymbol{x}(t); \boldsymbol{\theta}_k)dt)$ random variables, from which we have $\mathbb{E}(d\boldsymbol{R}(t)) = \boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})dt$ and $\mathrm{Cov}(d\boldsymbol{R}(t)) = \mathrm{diag}\{\boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})\}dt$. Under the Poisson assumption, we adopt the *diffusion approximation to Markov jump process* following the study Gillespie (2000) and then model $d\boldsymbol{R}(t)$ with Itô SDE, i.e.,

$$d\boldsymbol{R}(t) = \mathbb{E}(d\boldsymbol{R}(t)) + \{\mathrm{Cov}(d\boldsymbol{R}(t))\}^{\frac{1}{2}} d\boldsymbol{B}(t) = \boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})dt + \{\mathrm{diag}\{\boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})\}\}^{\frac{1}{2}} d\boldsymbol{B}(t),$$

where $d\boldsymbol{B}(t)$ is the increment of a K-dimensional standard Brownian motion. Given the reaction network structure specified by the stoichiometry matrix $\boldsymbol{C}$, the impact on the process dynamics becomes,

$$d\boldsymbol{x}(t) = \boldsymbol{C}d\boldsymbol{R}(t) = \boldsymbol{C}\boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})dt + \left\{\boldsymbol{C}\mathrm{diag}\{\boldsymbol{\omega}(\boldsymbol{x}(t); \boldsymbol{\theta})\}\boldsymbol{C}^\top\right\}^{\frac{1}{2}} d\boldsymbol{B}(t). \tag{1}$$

For both theoretical study and practical application purposes, the system is assumed to have a size parameter Ω (such as the volume of bioreactor). Then $s_j(t) = x_j(t)/\Omega$ represents the concentration of molecules of species j. At any time t, let $\boldsymbol{s}(t) = (s_1(t), s_2(t), \ldots, s_J(t))^\top = \Omega^{-1}\boldsymbol{x}(t)$ be the bioprocess state. And the propensity functions $\omega_k(\boldsymbol{x}(t); \boldsymbol{\theta}_k)$ for $k = 1, 2, \ldots, K$ can be written as

$$\omega_k(\boldsymbol{x}(t); \boldsymbol{\theta}_k) = \Omega v_k\left(\Omega^{-1}\boldsymbol{x}(t); \boldsymbol{\theta}_k\right) = \Omega v_k(\boldsymbol{s}(t); \boldsymbol{\theta}_k), \tag{2}$$

where v_k is the reaction rate associated with the k-th reaction, specified by the parameters $\boldsymbol{\theta}_k$ and depending on the current system state $\boldsymbol{s}(t)$. By plugging the relation between the propensity function and the reaction

rate (i.e., Equation (2)) into Equation (1), we get the state transition,

$$\begin{aligned} d\boldsymbol{s}(t) = \Omega^{-1} d\boldsymbol{x}(t) &= \boldsymbol{C}\boldsymbol{v}(\boldsymbol{s}(t);\boldsymbol{\theta})dt + \Omega^{-\frac{1}{2}} \left\{ \boldsymbol{C}\mathrm{diag}\{\boldsymbol{v}(\boldsymbol{s}(t);\boldsymbol{\theta})\}\boldsymbol{C}^\top \right\}^{\frac{1}{2}} d\boldsymbol{B}(t) \\ &\triangleq \boldsymbol{\mu}(\boldsymbol{s}(t);\boldsymbol{\theta})dt + \Omega^{-\frac{1}{2}} \{\boldsymbol{D}(\boldsymbol{s}(t);\boldsymbol{\theta})\}^{\frac{1}{2}} d\boldsymbol{B}(t), \end{aligned} \tag{3}$$

where $\boldsymbol{v}(\boldsymbol{s}(t);\boldsymbol{\theta}) = (v_1(\boldsymbol{s}(t);\boldsymbol{\theta}_1), v_2(\boldsymbol{s}(t);\boldsymbol{\theta}_2), \ldots, v_K(\boldsymbol{s}(t);\boldsymbol{\theta}_K))^\top$ is the reaction rate vector. Equation (3) also represents the *doubly stochastic property* of SRN, that is, both mean $\boldsymbol{\mu}(\boldsymbol{s}(t);\boldsymbol{\theta})$ and variance $\boldsymbol{D}(\boldsymbol{s}(t);\boldsymbol{\theta})$ are functions of the current system state $\boldsymbol{s}(t)$ and characterized by the parameters $\boldsymbol{\theta}$, while $\boldsymbol{s}(t)$ is a random state vector that changes over time and its evolution (i.e., $d\boldsymbol{s}(t)$) is characterized by $\boldsymbol{\mu}(\boldsymbol{s}(t);\boldsymbol{\theta})$ and $\boldsymbol{D}(\boldsymbol{s}(t);\boldsymbol{\theta})$.

(2) Partially observed state and heterogeneous data collection. The measures of partially observed state variables are often heterogeneous and subject to measurement errors. The observations for different observable state components are also asynchronous; see Figure 1(a). In particular, we represent all observation times of state as the time set, denoted by $\boldsymbol{T} = \{t_0, t_1, \ldots, t_H\}$, where $t_0 < t_1 < \cdots < t_H$, and the time intervals $\Delta t_h = t_{h+1} - t_h$ can be variable for $h = 0, 1, \ldots, H-1$. At each observation time t_h, we denote the set of observed components' subscripts of underlying state $\boldsymbol{s}$ by $\boldsymbol{J}_h$, i.e., $\boldsymbol{J}_h = \{j \in [J] : s_j \text{ is observed at time } t_h\}$ where $[J]$ represents $\{1, 2, \ldots, J\}$, and let $\boldsymbol{J}_y = \cup_{h=0}^{H} \boldsymbol{J}_h$ be the set of subscripts of the components that can be observed at certain times of experiments. The observations are denoted by $\boldsymbol{y}_h(t_h) \in \mathbb{R}^{M|\boldsymbol{J}_h|}$, where $|\boldsymbol{J}_h| \leq J$ is the cardinality of $\boldsymbol{J}_h$ representing the dimension of observed components of underlying state $\boldsymbol{s}$ at time t_h, and M is the batch size of experiments. Then, the observations at time t_h can be modeled as

$$\boldsymbol{y}_h(t_h) = \boldsymbol{G}_h \boldsymbol{s}(t_h) + \boldsymbol{\varepsilon}_h(t_h). \tag{4}$$

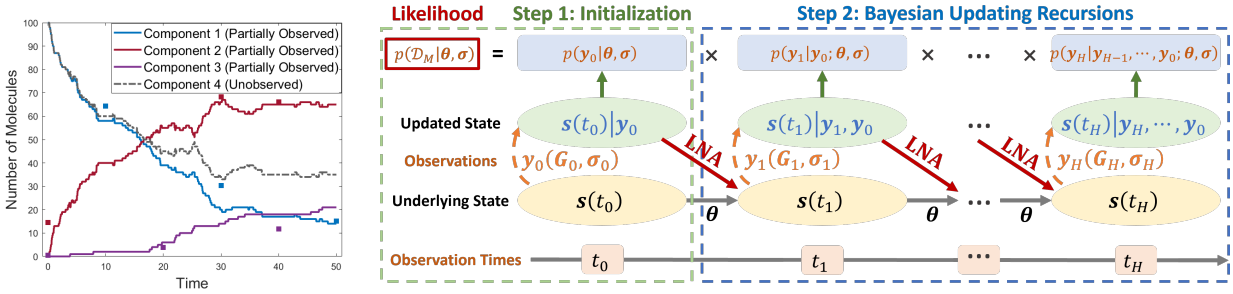
Suppose the measurement errors follow a multivariate Gaussian distribution $\boldsymbol{\varepsilon}_h(t_h) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_h)$, where $\boldsymbol{\Sigma}_h$ is a diagonal matrix with M vectors of $\boldsymbol{\sigma}_h$ on the main diagonal, and $\boldsymbol{\sigma}_h = \left\{\sigma_{jj} : j \in \boldsymbol{J}_h\right\}^\top$ is the vector of measurement error level at time t_h. Further, let $\boldsymbol{\sigma} = \left\{\sigma_{jj} : j \in \boldsymbol{J}_y\right\}^\top$ be the vector of measurement error level of all observed components. And $\boldsymbol{G}_h$ is a $M|\boldsymbol{J}_h|$-by-J constant matrix, mapping the entire J-dimensional vector of underlying state $\boldsymbol{s}(t_h)$ into the M batches of $|\boldsymbol{J}_h|$-dimensional vector containing only the counterpart of observed components at time t_h. Notice that the dimension $|\boldsymbol{J}_h|$ can change at different observation times accounting for the fact that the measures of partially observed state are asynchronous.

Given the observed data set denoted by $\mathscr{D}_M = \{\boldsymbol{y}_h(t_h)\}_{h=0}^{H}$, the model uncertainty is quantified by a posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{\sigma}|\mathscr{D}_M) \propto p(\boldsymbol{\theta})\, p(\boldsymbol{\sigma})\, p(\mathscr{D}_M|\boldsymbol{\theta}, \boldsymbol{\sigma})$. With the collection of new experiment data $\Delta\mathscr{D}$, the model uncertainty can be updated as $p(\boldsymbol{\theta}, \boldsymbol{\sigma}|\mathscr{D}_M \cup \Delta\mathscr{D}) \propto p(\boldsymbol{\theta}, \boldsymbol{\sigma}|\mathscr{D}_M)\, p(\Delta\mathscr{D}|\boldsymbol{\theta}, \boldsymbol{\sigma})$.

(3) Key challenges on Bayesian inference and summary of the proposed inference approach. Our focus in this paper is to develop a computationally efficient Bayesian inference approach on unknown model parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta}^N$ with emphasis on nonlinear $\boldsymbol{\mu}(\boldsymbol{s}(t);\boldsymbol{\theta})$ and $\boldsymbol{D}(\boldsymbol{s}(t);\boldsymbol{\theta})$ characterizing the regulation mechanisms of SRN as shown in (3), where $\boldsymbol{\Theta}^N \subset \mathbb{R}^N$ is the feasible parameter space. The first challenge is partially observed state subject to random measurement error. The observed components of system state $\boldsymbol{s}$ are recorded at limited discrete time points and the observation time points of each observable component may not be synchronized; see Figure 1(a). Moreover, there are often some components of state $\boldsymbol{s}$ unobservable. To tackle this challenge, we develop an interpretable *Bayesian updating LNA metamodel* on underlying state $\boldsymbol{s}(t)$ in Section 3 so that we can predict all components of $\boldsymbol{s}(t)$ at any time t.

Such a metamodel needs to have capability to characterize the dependence between components of $\boldsymbol{s}(t)$ and handle the doubly stochasticity of SRN. Luckily, we have a SDE model (3) representing the mechanism of state change. This brings us to the second challenge. On the one hand, the nonlinear drift and diffusion terms of the SDE (3) make solving it directly to get the metamodel of $\boldsymbol{s}(t)$ require time-consuming numerical integration methods. On the other hand, the regulation mechanism structure information from the SDE cannot be completely exploited which impacts on interpretability if we choose a black-box metamodel. To take full advantage of the structure information about the state transition provided by the SDE (3), and to

avoid the use of numerical integration to solve these SDEs, we place LNA priors on the dynamics of state $\boldsymbol{s}(t)$ to facilitate inference of model parameters $\boldsymbol{\theta}$. Under the LNA, the underlying process $\{\boldsymbol{s}(t) : t \geq 0\}$ follows a multivariate Gaussian distribution, combined with the assumption of linear Gaussian relation between each observation $\boldsymbol{y}_h(t_h)$ and underlying state value $\boldsymbol{s}(t_h)$ as shown in Equation (4) to give a tractable approximation to the likelihood of the observations $\{\boldsymbol{y}_h(t_h)\}_{h=0}^{H}$. The key to our approach is, to avoid a poor approximation to the true distribution of $\boldsymbol{s}(t)$ as t gets large, we reinitialize the LNA for each time interval $(t_h, t_{h+1}]$ using the derived posterior distribution of $\boldsymbol{s}(t_h)$ given $\boldsymbol{y}_h(t_h), \boldsymbol{y}_{h-1}(t_{h-1}), \ldots, \boldsymbol{y}_0(t_0)$; see Figure 1(b).



(a) Sample observations

(b) Bayesian updating linear noise approximation (LNA) metamodel

Figure 1: An illustration of (a) the partially observed state with measurement error; and (b) the proposed interpretable Bayesian updating LNA metamodel for enzymatic stochastic reaction network (SRN).

3 BAYESIAN UPDATING LINEAR NOISE APPROXIMATION (LNA) METAMODEL

In this section, we first utilize the LNA to approximate the SDE model (3) and then develop a Bayesian updating LNA metamodel to reduce the approximation error between the true solution to the SDE (3) and LNA model. The LNA divides the path $\{\boldsymbol{s}(t) : t \geq 0\}$ of the SDE (3) into a deterministic path $\{\bar{\boldsymbol{s}}(t) : t \geq 0\}$ and a stochastic perturbation $\{\boldsymbol{\xi}(t) : t \geq 0\}$, where the fluctuations in $\boldsymbol{s}(t)$ at any given time t are assumed to be of $O(\Omega^{-\frac{1}{2}})$; see Ferm et al. (2008) and Fearnhead et al. (2014) for a rigorous derivation and detailed discussion. Under this partition, through a Taylor expansion of the SDE (3) around $\bar{\boldsymbol{s}}(t)$ up to order $\Omega^{-\frac{1}{2}}$, we split the SDE (3) into one deterministic ODE with the solution $\bar{\boldsymbol{s}}(t)$ as shown in Equation (5),

$$d\bar{\boldsymbol{s}}(t) = \boldsymbol{\mu}(\bar{\boldsymbol{s}}(t); \boldsymbol{\theta})dt \tag{5}$$

with initial value $\bar{\boldsymbol{s}}(0)$, and one SDE with its solution $\boldsymbol{\xi}(t)$ following a Gaussian distribution for any fixed or Gaussian distributed initial condition on $\boldsymbol{\xi}(0)$, denoting by $\boldsymbol{\xi}(t) \sim \mathcal{N}(\boldsymbol{\varphi}(t), \boldsymbol{\Psi}(t))$. And its mean vector $\boldsymbol{\varphi}(t)$ and covariance matrix $\boldsymbol{\Psi}(t)$ for any $t \geq 0$ can be obtained by solving the ODEs in (6) and (7),

$$d\boldsymbol{\varphi}(t) = \nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t)}\boldsymbol{\varphi}(t)dt, \tag{6}$$

$$d\boldsymbol{\Psi}(t) = \left\{\boldsymbol{\Psi}(t)\left(\nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t)}\right)^{\top} + \nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t)}\boldsymbol{\Psi}(t) + \boldsymbol{D}(\bar{\boldsymbol{s}}(t); \boldsymbol{\theta})\right\}dt, \tag{7}$$

with initial values $\boldsymbol{\varphi}(0)$ and $\boldsymbol{\Psi}(0)$. Without loss of generality, in the following discussion, we simplify the notation and assume an unit system size $\Omega = 1$. Suppose the initial condition for the SDE (3) with $\Omega = 1$ is $\boldsymbol{s}(0) \sim \mathcal{N}(\boldsymbol{\alpha}^*(0), \boldsymbol{\beta}^*(0))$, then for arbitrary $\bar{\boldsymbol{s}}(0)$, we can set $\boldsymbol{\varphi}(0) = \boldsymbol{\alpha}^*(0) - \bar{\boldsymbol{s}}(0)$ and $\boldsymbol{\Psi}(0) = \boldsymbol{\beta}^*(0)$. Integrating the ODEs (5), (6), and (7) through time 0 to t provides the LNA

$$\boldsymbol{s}(t) \sim \mathcal{N}(\bar{\boldsymbol{s}}(t) + \boldsymbol{\varphi}(t), \boldsymbol{\Psi}(t)). \tag{8}$$

Under the LNA model (8) on the partially observed state $\boldsymbol{s}(t_h)$ with measurement error $\boldsymbol{\varepsilon}_h(t_h)$ as shown in (4) for $h = 0, 1, \ldots, H$, the likelihood of the observations $\mathscr{D}_M$ is tractable. In particular, the ODE components

of the LNA (i.e., Equations (5), (6), and (7)) are solved once over the entire time interval for given initial values. However, LNA can lead to a poor approximation to the true $\boldsymbol{s}(t)$, due to the approximation error between the true solution to the SDE (3) and the LNA (8) gradually accumulates as t gets large.

To tackle this issue, we construct the likelihood of the observations $\mathscr{D}_M$ through using the updated LNA model at each observation time point t_h with $h = 0, 1, \ldots, H$. In particular, given an estimate of model parameters $\boldsymbol{\theta}$ and measure error level $\boldsymbol{\sigma}$, we first set the LNA model (8) with the initial condition $\boldsymbol{s}(t_0) \sim \mathscr{N}(\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0), \boldsymbol{\Psi}(t_0))$ as a prior, and then the observations $\boldsymbol{y}_h(t_h) \in \mathscr{D}_M$ are used to *sequentially* update the prior on $\boldsymbol{s}(t_h)$ for each t_h with the procedure shown in Figure 1(b). Therefore, we can approximate the distribution of $\boldsymbol{y}_h(t_h)$ given all observations up to time t_h and obtain the likelihood. The detailed procedure is summarized in the following three steps.

Step 1: At the initial observation time point t_0, given the prior $\boldsymbol{s}(t_0) \sim \mathscr{N}(\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0), \boldsymbol{\Psi}(t_0))$ and the observational uncertainty (4), we can directly have

$$\boldsymbol{y}_0(t_0)|\boldsymbol{\sigma} \sim \mathscr{N}\left(\boldsymbol{G}_0\left\{\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0)\right\}, \boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top + \boldsymbol{\Sigma}_0\right). \tag{9}$$

Combining the LNA prior of $\boldsymbol{s}(t_0)$ with (9), we obtain the joint distribution of $\boldsymbol{s}(t_0)$ and $\boldsymbol{y}_0(t_0)$ as

$$\begin{pmatrix}\boldsymbol{s}(t_0)\\ \boldsymbol{y}_0(t_0)\end{pmatrix}\Bigg|\boldsymbol{\sigma} \sim \mathscr{N}\left\{\begin{pmatrix}\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0)\\ \boldsymbol{G}_0\left\{\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0)\right\}\end{pmatrix}, \begin{pmatrix}\boldsymbol{\Psi}(t_0) & \boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top\\ \boldsymbol{G}_0\boldsymbol{\Psi}(t_0) & \boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top + \boldsymbol{\Sigma}_0\end{pmatrix}\right\}.$$

By applying the conditional distribution properties of multivariate Gaussian distribution, the posterior distribution of $\boldsymbol{s}(t_0)$ is updated based on the observation $\boldsymbol{y}_0(t_0)$, i.e.,

$$\boldsymbol{s}(t_0)|\boldsymbol{y}_0(t_0); \boldsymbol{\sigma} \sim \mathscr{N}(\boldsymbol{\alpha}(t_0), \boldsymbol{\beta}(t_0)),$$

where

$$\boldsymbol{\alpha}(t_0) = \bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0) + \boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top\left(\boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top + \boldsymbol{\Sigma}_0\right)^{-1}\left(\boldsymbol{y}_0(t_0) - \boldsymbol{G}_0\left\{\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0)\right\}\right), \tag{10}$$

$$\boldsymbol{\beta}(t_0) = \boldsymbol{\Psi}(t_0) - \boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top\left(\boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top + \boldsymbol{\Sigma}_0\right)^{-1}\boldsymbol{G}_0\boldsymbol{\Psi}(t_0). \tag{11}$$

Step 2: For the subsequent observation time points $t_1, t_2, \ldots, t_H$, we apply the idea of Kalman filter to sequentially update the LNA prior for $\boldsymbol{s}(t_{h+1})$ and calculate the approximate $p(\boldsymbol{y}_{h+1}(t_{h+1})|\boldsymbol{y}_h(t_h), \ldots, \boldsymbol{y}_0(t_0); \boldsymbol{\theta}, \boldsymbol{\sigma})$ recursively for $h = 0, 1, \ldots, H-1$. Specifically, we first reinitialize the initial values of the ODEs (5) and (7) to the posterior mean and covariance of $\boldsymbol{s}(t_h)$ respectively. That is, set $\bar{\boldsymbol{s}}(t_h) = \boldsymbol{\alpha}(t_h)$ and $\boldsymbol{\Psi}(t_h) = \boldsymbol{\beta}(t_h)$. We let $\boldsymbol{\varphi}(t_h) = 0$ as $\boldsymbol{\varphi}(t_k) = 0$ for all $k \geq h$ according to the ODE (6). By integrating the ODEs (5) and (7) through time t_h to t_{h+1}, we obtain $\bar{\boldsymbol{s}}(t_{h+1})$ and $\boldsymbol{\Psi}(t_{h+1})$. In practice we work with their discretized versions, given by the Euler method,

$$\bar{\boldsymbol{s}}(t_{h+1}) = \bar{\boldsymbol{s}}(t_h) + \boldsymbol{\mu}(\bar{\boldsymbol{s}}(t_h); \boldsymbol{\theta})\Delta t_h, \tag{12}$$

$$\boldsymbol{\Psi}(t_{h+1}) = \boldsymbol{\Psi}(t_h) + \left\{\boldsymbol{\Psi}(t_h)\left(\nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t_h)}\right)^\top + \nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t_h)}\boldsymbol{\Psi}(t_h) + \boldsymbol{D}(\bar{\boldsymbol{s}}(t_h); \boldsymbol{\theta})\right\}\Delta t_h. \tag{13}$$

As $\Delta t_h = t_{h+1} - t_h$ is often too large to be used as a time step in (12) and (13), we introduce $\Delta z_h = \Delta t_h / I_h$ for some positive integer $I_h \geq 1$. By choosing I_h to be sufficiently large, we can ensure the discretization error associated with the Euler method is arbitrarily small. That is, to compute $\bar{\boldsymbol{s}}(t_{h+1})$ and $\boldsymbol{\Psi}(t_{h+1})$ more accurately, we recursively calculate the following equations for $i = 0, 1, \ldots, I_h - 1$,

$$\bar{\boldsymbol{s}}(t_h + (i+1)\Delta z_h) = \bar{\boldsymbol{s}}(t_h + i\Delta z_h) + \boldsymbol{\mu}(\bar{\boldsymbol{s}}(t_h + i\Delta z_h); \boldsymbol{\theta})\Delta z_h, \tag{14}$$

$$\boldsymbol{\Psi}(t_h + (i+1)\Delta z_h) = \boldsymbol{\Psi}(t_h + i\Delta z_h) + \left\{\boldsymbol{\Psi}(t_h + i\Delta z_h)\left(\nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s}; \boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t_h + i\Delta z_h)}\right)^\top +$$

$$\nabla_{\boldsymbol{s}}\boldsymbol{\mu}(\boldsymbol{s};\boldsymbol{\theta})|_{\boldsymbol{s}=\bar{\boldsymbol{s}}(t_h+i\Delta z_h)}\boldsymbol{\Psi}(t_h+i\Delta z_h)+\boldsymbol{D}(\bar{\boldsymbol{s}}(t_h+i\Delta z_h);\boldsymbol{\theta})\big\}\Delta z_h. \tag{15}$$

Therefore, we get the updated LNA prior on $\boldsymbol{s}(t_{h+1})$ by applying (8), i.e.,

$$\boldsymbol{s}(t_{h+1})|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}\sim\mathscr{N}\left(\bar{\boldsymbol{s}}(t_{h+1}),\boldsymbol{\Psi}(t_{h+1})\right). \tag{16}$$

Here, LNA gives us a Gaussian approximation to the transition density from $\boldsymbol{s}(t_h)|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}$ to $\boldsymbol{s}(t_{h+1})|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}$. Then, based on the model of measurement uncertainty or error in (4), we get a one-step forecast of the observation $\boldsymbol{y}_{h+1}(t_{h+1})$ as

$$\boldsymbol{y}_{h+1}(t_{h+1})|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}\sim\mathscr{N}\left(\boldsymbol{G}_{h+1}\bar{\boldsymbol{s}}(t_{h+1}),\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\right). \tag{17}$$

Combining the distributions (16) and (17), we obtain the joint distribution as

$$\begin{pmatrix}\boldsymbol{s}(t_{h+1})\\ \boldsymbol{y}_{h+1}(t_{h+1})\end{pmatrix}\Bigg|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}\sim\mathscr{N}\left\{\begin{pmatrix}\bar{\boldsymbol{s}}(t_{h+1})\\ \boldsymbol{G}_{h+1}\bar{\boldsymbol{s}}(t_{h+1})\end{pmatrix},\begin{pmatrix}\boldsymbol{\Psi}(t_{h+1}) & \boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top\\ \boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1}) & \boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\end{pmatrix}\right\}.$$

Thus, the posterior distribution of $\boldsymbol{s}(t_{h+1})$ becomes

$$\boldsymbol{s}(t_{h+1})|\boldsymbol{y}_{h+1}(t_{h+1}),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}\sim\mathscr{N}\left(\boldsymbol{\alpha}(t_{h+1}),\boldsymbol{\beta}(t_{h+1})\right),$$

where

$$\boldsymbol{\alpha}(t_{h+1})=\bar{\boldsymbol{s}}(t_{h+1})+\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top\left(\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\right)^{-1}\left(\boldsymbol{y}_{h+1}(t_{h+1})-\boldsymbol{G}_{h+1}\bar{\boldsymbol{s}}(t_{h+1})\right), \tag{18}$$

$$\boldsymbol{\beta}(t_{h+1})=\boldsymbol{\Psi}(t_{h+1})-\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top\left(\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\right)^{-1}\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1}). \tag{19}$$

Step 3: From the distributions (9) and (17) for $h=0,1,\ldots,H-1$, the likelihood of the observations $\mathscr{D}_M$ can be calculated by the following decomposition,

$$p\left(\boldsymbol{y}_0(t_0),\boldsymbol{y}_1(t_1),\ldots,\boldsymbol{y}_H(t_H)|\boldsymbol{\theta},\boldsymbol{\sigma}\right)=p\left(\boldsymbol{y}_0(t_0)|\boldsymbol{\theta},\boldsymbol{\sigma}\right)\prod_{h=0}^{H-1}p\left(\boldsymbol{y}_{h+1}(t_{h+1})|\boldsymbol{y}_h(t_h),\ldots,\boldsymbol{y}_0(t_0);\boldsymbol{\theta},\boldsymbol{\sigma}\right).$$

4 BAYESIAN ANALYSIS AND ALGORITHM DEVELOPMENT

In this section, we simultaneously infer the model parameters $\boldsymbol{\theta}$ and the measurement error level $\boldsymbol{\sigma}$ from the observations $\mathscr{D}_M$. By applying the Bayes' rule, we have the joint posterior distribution of $\boldsymbol{\theta}$ and $\boldsymbol{\sigma}$,

$$\begin{aligned}p\left(\boldsymbol{\theta},\boldsymbol{\sigma}|\mathscr{D}_M\right)\propto p\left(\boldsymbol{\theta}\right)p\left(\boldsymbol{\sigma}\right)\exp\Bigg\{-\frac{1}{2}\Bigg[&M|\boldsymbol{J}_0|\log(2\pi)+\log\left|\boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top+\boldsymbol{\Sigma}_0\right|\\&+\left(\boldsymbol{y}_0(t_0)-\boldsymbol{G}_0\left\{\bar{\boldsymbol{s}}(t_0)+\boldsymbol{\varphi}(t_0)\right\}\right)^\top\left(\boldsymbol{G}_0\boldsymbol{\Psi}(t_0)\boldsymbol{G}_0^\top+\boldsymbol{\Sigma}_0\right)^{-1}\left(\boldsymbol{y}_0(t_0)-\boldsymbol{G}_0\left\{\bar{\boldsymbol{s}}(t_0)+\boldsymbol{\varphi}(t_0)\right\}\right)\Bigg]\\&-\frac{1}{2}\sum_{h=0}^{H-1}\Bigg[M|\boldsymbol{J}_{h+1}|\log(2\pi)+\log\left|\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\right|\\&+\left(\boldsymbol{y}_{h+1}(t_{h+1})-\boldsymbol{G}_{h+1}\bar{\boldsymbol{s}}(t_{h+1})\right)^\top\left(\boldsymbol{G}_{h+1}\boldsymbol{\Psi}(t_{h+1})\boldsymbol{G}_{h+1}^\top+\boldsymbol{\Sigma}_{h+1}\right)^{-1}\left(\boldsymbol{y}_{h+1}(t_{h+1})-\boldsymbol{G}_{h+1}\bar{\boldsymbol{s}}(t_{h+1})\right)\Bigg]\Bigg\},\end{aligned} \tag{20}$$

where $p\left(\boldsymbol{\theta}\right)$ and $p\left(\boldsymbol{\sigma}\right)$ are the priors for $\boldsymbol{\theta}$ and $\boldsymbol{\sigma}$ respectively. By utilizing the joint posterior distribution $p\left(\boldsymbol{\theta},\boldsymbol{\sigma}|\mathscr{D}_M\right)$ in (20), we further develop a MALA procedure to efficiently generate posterior samples of the L-dimensional parameters $\boldsymbol{\eta}\equiv\left(\boldsymbol{\theta}^\top,\boldsymbol{\sigma}^\top\right)^\top$ where $L=N+|\boldsymbol{J}_y|$.

By utilizing the gradients of posterior (20), MALA generates more promising candidate samples at the parameter space with higher posterior probability. It improves the mixing of classic MCMC algorithm through utilizing a combination of two mechanisms, i.e., Langevin diffusion and Metropolis-Hastings step. Langevin diffusion is originally a gradient descent of a potential function (representing a force field in physics) plus a Brownian motion term accounting for thermodynamics. To overcome the limitation of random walk-based search strategies used in classic MCMC, we leverage on the information provided by the closed form posterior distribution $p(\boldsymbol{\eta}|\mathscr{D}_M)$ and use Langevin diffusion to develop a more efficient posterior sampling approach. We construct a continuous-time stochastic process characterizing the Langevin diffusion-based posterior search. Specifically, we consider the following (overdamped) Langevin diffusion

$$d\boldsymbol{\eta}(\tau) = \nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)\,|_{\boldsymbol{\eta}=\boldsymbol{\eta}(\tau)} d\tau + \sqrt{2} d\boldsymbol{W}(\tau) \tag{21}$$

driven by the time derivative of an L-dimensional standard Brownian motion (i.e., $d\boldsymbol{W}(\tau)$). It speeds up the MCMC convergence through drifting the search with the gradient of the target log-posterior distribution (i.e., $\log p(\boldsymbol{\eta}|\mathscr{D}_M)$), which drives the random walk towards the parameter region with high posterior probability.

To numerically solve Equation (21) and generate posterior samples from $p(\boldsymbol{\eta}|\mathscr{D}_M)$, the Euler-Maruyama approximation (Kloeden and Platen 1992) is used to obtain the discretized Langevin diffusion with a step size $\Delta\tau > 0$,

$$\boldsymbol{\eta}(\tau+1) := \boldsymbol{\eta}(\tau) + \nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)\,|_{\boldsymbol{\eta}=\boldsymbol{\eta}(\tau)} \Delta\tau + \sqrt{2}\Delta\boldsymbol{W}(\tau), \tag{22}$$

where each $\Delta\boldsymbol{W}(\tau) \in \mathbb{R}^L$ is a Gaussian random vector with mean zero and covariance $\text{diag}\{\Delta\tau\} \in \mathbb{R}^{L\times L}$. The gradient of the log-posterior

$$\nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M) = \left(\left\{ \frac{\partial \log p(\boldsymbol{\theta},\boldsymbol{\sigma}|\mathscr{D}_M)}{\partial \theta_n}, n \in [N] \right\}, \left\{ \frac{\partial \log p(\boldsymbol{\theta},\boldsymbol{\sigma}|\mathscr{D}_M)}{\partial \sigma_{jj}}, j \in \boldsymbol{J}_y \right\} \right)^\top$$

is tractable from Equation (20). In particular, we provide a recursive procedure in Algorithm 1 to compute $p(\boldsymbol{\eta}|\mathscr{D}_M)$ and $\nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)$.

Algorithm 1: Computing $p(\boldsymbol{\eta}|\mathscr{D}_M)$ and $\nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)$.

Input: The priors $p(\boldsymbol{\theta})$ and $p(\boldsymbol{\sigma})$, observations $\mathscr{D}_M = \{\boldsymbol{y}_h(t_h)\}_{h=0}^H$, ODE initial values $\bar{\boldsymbol{s}}(t_0)$, $\boldsymbol{\varphi}(t_0)$ and $\boldsymbol{\Psi}(t_0)$, constant matrices $\boldsymbol{G}_h$, and appropriate positive integers I_h for $h = 0, 1, \ldots, H-1$.

Output: $p(\boldsymbol{\eta}|\mathscr{D}_M)$ and $\nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)$.

1. Calculate $\boldsymbol{\alpha}(t_0)$ and $\boldsymbol{\beta}(t_0)$ by applying Equations (10) and (11);

2. Calculate $\partial\boldsymbol{\alpha}(t_0)/\partial\theta_n$, $\partial\boldsymbol{\beta}(t_0)/\partial\theta_n$ for $n \in [N]$, and $\partial\boldsymbol{\alpha}(t_0)/\partial\sigma_{jj}$, $\partial\boldsymbol{\beta}(t_0)/\partial\sigma_{jj}$ for $j \in \boldsymbol{J}_y$;

for $h = 0, 1, \ldots, H-1$ **do**

  for $i = 0, 1, \ldots, I_h - 1$ **do**

    3. Calculate $\bar{\boldsymbol{s}}(t_h + (i+1)\Delta z_h)$ and $\boldsymbol{\Psi}(t_h + (i+1)\Delta z_h)$ by applying Equations (14) and (15);

    4. Calculate $\partial\bar{\boldsymbol{s}}(t_h + (i+1)\Delta z_h)/\partial\theta_n$, $\partial\boldsymbol{\Psi}(t_h + (i+1)\Delta z_h)/\partial\theta_n$ for $n \in [N]$, and $\partial\bar{\boldsymbol{s}}(t_h + (i+1)\Delta z_h)/\partial\sigma_{jj}$, $\partial\boldsymbol{\Psi}(t_h + (i+1)\Delta z_h)/\partial\sigma_{jj}$ for $j \in \boldsymbol{J}_y$;

  5. Calculate $\boldsymbol{\alpha}(t_{h+1})$ and $\boldsymbol{\beta}(t_{h+1})$ by applying Equations (18) and (19);

  6. Calculate $\partial\boldsymbol{\alpha}(t_{h+1})/\partial\theta_n$, $\partial\boldsymbol{\beta}(t_{h+1})/\partial\theta_n$ for $n \in [N]$, and $\partial\boldsymbol{\alpha}(t_{h+1})/\partial\sigma_{jj}$, $\partial\boldsymbol{\beta}(t_{h+1})/\partial\sigma_{jj}$ for $j \in \boldsymbol{J}_y$;

7. Return $p(\boldsymbol{\eta}|\mathscr{D}_M)$ by applying Equation (20), and $\nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)$ by calculating $\partial \log p(\boldsymbol{\eta}|\mathscr{D}_M)/\partial\theta_n$ for $n \in [N]$ and $\partial \log p(\boldsymbol{\eta}|\mathscr{D}_M)/\partial\sigma_{jj}$ for $j \in \boldsymbol{J}_y$.

To correct the bias in the stationary distribution induced by the discretization used in the update rule (22), a Metropolis-Hastings step is incorporated for simulating the Langevin diffusion (21). Specifically, we consider the update rule (22) and define a proposal distribution to generate a new MCMC posterior sample $\tilde{\boldsymbol{\eta}}(\tau+1)$,

$$\tilde{\boldsymbol{\eta}}(\tau+1) := \boldsymbol{\eta}(\tau) + \nabla_{\boldsymbol{\eta}} \log p(\boldsymbol{\eta}|\mathscr{D}_M)\,|_{\boldsymbol{\eta}=\boldsymbol{\eta}(\tau)} \Delta\tau + \sqrt{2}\Delta\boldsymbol{W}(\tau). \tag{23}$$

Thus, the MCMC conditional sampling distribution $\tilde{\boldsymbol{\eta}}(\tau+1)|\boldsymbol{\eta}(\tau)$ is Gaussian distributed with mean $\boldsymbol{\eta}(\tau)+\nabla_{\boldsymbol{\eta}}\log p(\boldsymbol{\eta}|\mathscr{D}_M)|_{\boldsymbol{\eta}=\boldsymbol{\eta}(\tau)}\Delta\tau$ and covariance $\text{diag}\{2\Delta\tau\}\in\mathbb{R}^{L\times L}$. Then, the candidate sample $\tilde{\boldsymbol{\eta}}(\tau+1)$ generated from this proposal is accepted with the ratio,

$$\gamma_{\text{acc}} := \min\left\{1, \frac{p(\tilde{\boldsymbol{\eta}}(\tau+1)|\mathscr{D}_M)\, q_{\text{trans}}\left(\boldsymbol{\eta}(\tau)\middle|\tilde{\boldsymbol{\eta}}(\tau+1)\right)}{p(\boldsymbol{\eta}(\tau)|\mathscr{D}_M)\, q_{\text{trans}}\left(\tilde{\boldsymbol{\eta}}(\tau+1)\middle|\boldsymbol{\eta}(\tau)\right)}\right\}, \tag{24}$$

where the proposal distribution $q_{\text{trans}}(\boldsymbol{\eta}'|\boldsymbol{\eta}) \propto \exp\left\{-\frac{1}{4\Delta\tau}\left|\left|\boldsymbol{\eta}'-\boldsymbol{\eta}-\nabla_{\boldsymbol{\eta}}\log p(\boldsymbol{\eta}|\mathscr{D}_M)\Delta\tau\right|\right|^2\right\}$ ($||\cdot||$ denotes the Euclidean norm) is the transition density from $\boldsymbol{\eta}$ to $\boldsymbol{\eta}'$ obtained from Equation (23).

In sum, we provide the MALA procedure in Algorithm 2 to generate posterior samples for the enzymatic SRN mechanistic model parameters $\boldsymbol{\theta}$ and the measurement error level $\boldsymbol{\sigma}$ together. Within each τ-th iteration of MALA joint posterior sampler, given the previous sample $\boldsymbol{\eta}(\tau)$, we compute and generate one proposal sample $\tilde{\boldsymbol{\eta}}(\tau+1)$ from the discretized Langevin diffusion (23), and accept it with the Metropolis-Hastings ratio (24). By repeating this procedure, $\boldsymbol{\theta}$ and $\boldsymbol{\sigma}$ are updated together with a joint gradient at each iteration, and we thus get samples $\boldsymbol{\eta}(\tau) = \left(\boldsymbol{\theta}^\top(\tau), \boldsymbol{\sigma}^\top(\tau)\right)^\top$ with $\tau = 1,2,\ldots,T_0+(B-1)\delta$. To reduce the initial bias and correlations between consecutive samples, we discard an appropriate burn-in period for convergence (i.e., the first T_0 samples) and then keep one for every δ samples. Consequently, we obtain the posterior samples $\boldsymbol{\eta}^{(b)} \sim p(\boldsymbol{\eta}|\mathscr{D}_M)$ with $b = 1,2,\ldots,B$.

Algorithm 2: MALA joint posterior sampler for SRN.

Input: The priors $p(\boldsymbol{\theta})$ and $p(\boldsymbol{\sigma})$, step size $\Delta\tau$, posterior sample size B, initial warm-up length T_0, and an appropriate integer δ to reduce sample correlation.

Output: Posterior samples $\boldsymbol{\eta}^{(b)} \sim p(\boldsymbol{\eta}|\mathscr{D}_M)$ with $b = 1,2,\ldots,B$.

1. Set the initial values $\boldsymbol{\eta}(0) := \left(\boldsymbol{\theta}^\top(0), \boldsymbol{\sigma}^\top(0)\right)^\top$ by sampling from the priors;

for $\tau = 0,1,\ldots,T_0+(B-1)\delta$ **do**

- **2.** Calculate $p(\boldsymbol{\eta}(\tau)|\mathscr{D}_M)$ and $\nabla_{\boldsymbol{\eta}}\log p(\boldsymbol{\eta}|\mathscr{D}_M)|_{\boldsymbol{\eta}=\boldsymbol{\eta}(\tau)}$ by calling Algorithm 1;
- **3.** Generate a proposal $\tilde{\boldsymbol{\eta}}(\tau+1)$ by applying Equation (23);
- **4.** Calculate the acceptance ratio γ_{acc} by applying Equation (24);
- **5.** Draw u from the continuous uniform distribution $U(0,1)$;
- **if** $u \le \gamma_{\text{acc}}$ **then**
  - **6.** The proposal $\tilde{\boldsymbol{\eta}}(\tau+1)$ is accepted, and set $\boldsymbol{\eta}(\tau+1) := \tilde{\boldsymbol{\eta}}(\tau+1)$;
- **else if** $u > \gamma_{\text{acc}}$ **then**
  - **7.** The proposal $\tilde{\boldsymbol{\eta}}(\tau+1)$ is rejected, and set $\boldsymbol{\eta}(\tau+1) := \boldsymbol{\eta}(\tau)$;

8. Return posterior samples $\boldsymbol{\eta}^{(b)} := \boldsymbol{\eta}(T_0+(b-1)\delta+1)$ for $b = 1,2,\ldots,B$.

Remark 1 From Equation (23), the support of the posterior samples $\boldsymbol{\eta}^{(b)}$ generated by Algorithm 2 is the entire L-dimensional real space $\mathbb{R}^L$. But in most real-word cases including SRN, the feasible space of $\boldsymbol{\eta}$ is restricted, meaning it can be a subset of $\mathbb{R}^L$. For instance, some biological parameters such as rates should be ensured positivity, while some parameters such as probabilities or bioavailability should be between 0 and 1 (Prague et al. 2013). Reparametrization of the system allows us to take these constraints into account. Specifically, we can introduce one-to-one functions $f_l(\cdot)$ for $l = 1,2,\ldots,L$, and define transformed parameters $\eta_l^{\text{trans}} = f_l(\eta_l)$. For instance, $f_l(\cdot)$ can be logarithmic functions to transform the support from the positive space to the real space, or inverse Logistic functions to transform the support from the interval $[0,1]$ to the real space. Then we can perform Algorithm 2 on the transformed $\boldsymbol{\eta}^{\text{trans}} = (\eta_1^{\text{trans}}, \eta_2^{\text{trans}}, \ldots, \eta_L^{\text{trans}})^\top$.

5 EMPIRICAL STUDY

In this section, we use a representative example of SRN, i.e., Michaelis-Menten enzyme kinetics (Rao and Arkin 2003), to assess the empirical performance of the proposed Bayesian inference approach. In specific, we consider the Michaelis-Menten enzyme kinetic model involving four biochemical species, i.e., Enzyme, Substrate, Complex, and Product. It describes the catalytic conversion of a substrate into a product via an enzymatic reaction involving enzyme, represented by the following three chemical reactions,

$$\begin{array}{l}\text{Reaction 1 : Enzyme} + \text{Substrate} \longrightarrow \text{Complex},\\ \text{Reaction 2 : Complex} \longrightarrow \text{Enzyme} + \text{Substrate},\\ \text{Reaction 3 : Complex} \longrightarrow \text{Enzyme} + \text{Product},\end{array} \qquad \text{with} \quad \boldsymbol{C} = \begin{pmatrix} -1 & 1 & 1 \\ -1 & 1 & 0 \\ 1 & -1 & -1 \\ 0 & 0 & 1 \end{pmatrix}.$$

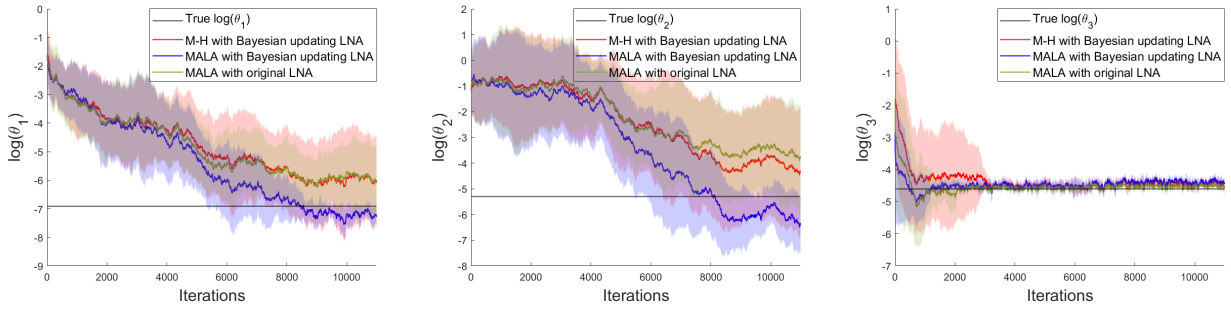
In particular, let $\boldsymbol{s}(t) = (s_1(t), s_2(t), s_3(t), s_4(t))^\top$ denote the system state vector at any time t, where $s_1(t)$, $s_2(t)$, $s_3(t)$, and $s_4(t)$ are the respective concentration of Enzyme, Substrate, Complex, and Product. The stoichiometry matrix $\boldsymbol{C}$ associated with the system can be obtained from the above three reaction equations, and the associated reaction rate vector is $\boldsymbol{v}(\boldsymbol{s}(t);\boldsymbol{\theta}) = (\theta_1 s_1(t) s_2(t), \theta_2 s_3(t), \theta_3 s_3(t))^\top$. Our goal is to perform Bayesian inference for the vector of the unknown kinetic rate parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)^\top$.

We simulate a synthetic dataset for 80 seconds (i.e., on the time interval $[0, 80]$ seconds) using the Gillespie algorithm (Gillespie 1977) to ensure exact simulation with the true parameters $\boldsymbol{\theta}^{\text{true}} = (0.001, 0.005, 0.01)^\top$, and the initial states $\boldsymbol{s}(0) = (45, 39, 55, 6)^\top$. These initial values are obtained after running the process for a short time from some arbitrarily chosen population levels. And we create a challenging data-poor scenario for model inference by assuming incomplete and noisy observations. Specifically, we consider one batch size (i.e., $M = 1$), and discard observations on the Enzyme, Substrate, and Product levels, and only the Complex level is observed every Δt seconds from $t_0 = 0$ to $t_H = 80$ ($H+1$ observation time points in total), so that $\boldsymbol{J}_h = \{3\}$ and $\boldsymbol{G}_h = (0, 0, 1, 0)$ for any $h = 0, 1, \ldots, H$. And we assume that there is homogeneous additive Gaussian measurement error, i.e., $\varepsilon(t_h) \sim \mathcal{N}(0, \sigma)$ where $\sigma = 4$; that is, the error standard deviation is two Complex molecules. The inferred parameter vector is $\boldsymbol{\eta} \equiv (\boldsymbol{\theta}^\top, \sigma)^\top$.

We assess the performance of the proposed joint posterior sampler under model uncertainty induced with the different data sizes, i.e., $H = 4, 8, 16$ ($\Delta t = 20, 10, 5$ seconds correspondingly). To study the effect of additional gradient information and Bayesian updating step, the MALA with Bayesian updating LNA is compared to other candidate approaches, including (1) classic Metropolis-Hastings algorithm (M-H) with Bayesian updating LNA, and (2) MALA with original LNA (without Bayesian update), in terms of convergence behavior of posterior sampling. Since the support of the parameters is the positive space, we first need to use the logarithmic function to transform it to the real space. That is, we set $\log(\boldsymbol{\eta}) = (\log(\theta_1), \log(\theta_2), \log(\theta_3), \log(\sigma))^\top$ and apply three algorithms to $\log(\boldsymbol{\eta})$. For both LNA metamodels, we set $\bar{\boldsymbol{s}}(t_0) + \boldsymbol{\varphi}(t_0) = (50, 40, 60, 10)^\top$, $\boldsymbol{\Psi}(t_0)$ as a 4-by-4 identity matrix, and $I_h = 2000, 1000, 500$ for $\Delta t = 20, 10, 5$ respectively to make a small $\Delta z_h = 0.01$ for any $h = 0, 1, \ldots, H$. The priors of the parameters are set as $\theta_k \sim U(0, 1)$ for $k = 1, 2, 3$, and $\sigma \sim U(0, 25)$, to consider a difficult situation without strong prior information. The results are estimated based on 10 macro-replications.

First, we compare the convergence speed of three algorithms. For MALA with Bayesian updating LNA and with original LNA (without Bayesian update), we set the step size $\Delta\tau = 0.001$. Correspondingly, to show that MALA improves the mixing of MCMC, for M-H with Bayesian updating LNA, we set its proposal distribution to be Gaussian distributed with the current sample as mean and $\text{diag}\{2\Delta\tau\} = \text{diag}\{0.002\}$ as covariance. Figure 2 shows the mean convergence trends of the three algorithms for the three log-kinetic rate parameters (with 95% confidence intervals (CIs) across 10 macro-replications) under the data size $H = 16$ ($\Delta t = 5$). The black line represents the true log-parameters. By comparing the widths of the CIs as iterations progress, we observe that MALA shows a significant improvement in the convergence speed of the log-kinetic rate parameters inference over M-H, while the Bayesian updating step reduces the approximation error accumulation over time of original LNA, providing a more accurate likelihood approximation. It

demonstrates that by sufficiently leveraging on the likelihood and its gradient information provided by a moderate size of observations, MALA posterior sampling based on the likelihood approximated by the Bayesian updating LNA metamodel converges quickly towards the true log-parameters.



(a) $\log(\theta_1)$ (b) $\log(\theta_2)$ (c) $\log(\theta_3)$

Figure 2: The convergence trends of (1) MALA with original LNA, (2) MALA with Bayesian updating LNA, and (3) M-H with Bayesian updating LNA (with 95% CIs) when the data size $H = 16$ ($\Delta t = 5$).

Then, we study the root mean square error (RMSE) of model parameter estimation to assess the convergence results of three algorithms under three different data sizes. Basically, the RMSE measures the differences between the true and the estimated log-parameters based on B posterior samples, i.e., RMSE $= \sqrt{\frac{1}{B}\sum_{b=1}^{B} |\log(\eta_l^{\text{true}}) - \{\log(\eta_l)\}^{(b)}|^2}$ for $l = 1,2,3,4$. We set the initial warm-up length $T_0 = 10000$ to reduce the initial bias, the posterior sample size $B = 100$, and an appropriate integer $\delta = 10$ to reduce sample correlation for three algorithms. We summarize the 95% CIs obtained by using 10 macro-replications of the RMSEs for the four log-parameters inferred by the three algorithms in Table 1. As it shows, for the four log-parameters except $\log(\theta_3)$ inferred under the data size $H = 8,16$, the RMSEs of MALA with Bayesian updating LNA decrease more significantly than those of MALA with original LNA as the data size increases, demonstrating the major benefit induced by the Bayesian updating LNA metamodel. That means it refines the approximation of the likelihood as more observations available to update the original LNA model. The exception of $\log(\theta_3)$ is due to that, θ_3 is the kinetic rate parameter associated with the Reaction 3, whose reactant (i.e., Complex) is the only observable component of this system, the data size of $H = 16$ is thus sufficient to provide relatively accurate likelihood information based on the original LNA metamodel, while the Bayesian updating step improves accuracy of likelihood approximation when the data size $H = 8$. With the help of MALA, $\log(\theta_3)$ converges close to the true value when $H = 8,16$. Additionally, for $\log(\theta_1)$ and $\log(\theta_2)$ under all data sizes, based on the likelihood approximated by the Bayesian updating LNA metamodel, MALA performs better than M-H in terms of both RMSEs and their confidence half-widths, meaning that MALA converges faster than M-H; while for $\log(\theta_3)$ and $\log(\sigma)$, the performance of the two algorithms is similar. This is because both two algorithms have converged before $T_0 = 10000$.

Table 1: The RMSEs between the estimated and the true log-parameters (with 95% CIs).

	MALA with Bayesian updating LNA			M-H with Bayesian updating LNA			MALA with original LNA		
Data size	$H=4$ ($\Delta t=20$)	$H=8$ ($\Delta t=10$)	$H=16$ ($\Delta t=5$)	$H=4$ ($\Delta t=20$)	$H=8$ ($\Delta t=10$)	$H=16$ ($\Delta t=5$)	$H=4$ ($\Delta t=20$)	$H=8$ ($\Delta t=10$)	$H=16$ ($\Delta t=5$)
$\log(\theta_1)$	1.79 ± 0.73	1.27 ± 0.92	0.48 ± 0.08	2.56 ± 1.09	1.80 ± 1.07	1.48 ± 1.37	1.82 ± 0.74	1.72 ± 0.81	1.41 ± 0.94
$\log(\theta_2)$	2.76 ± 1.37	2.12 ± 1.31	1.32 ± 0.58	3.29 ± 1.82	2.87 ± 1.49	2.50 ± 1.61	2.80 ± 1.38	2.73 ± 1.36	2.63 ± 1.19
$\log(\theta_3)$	1.73 ± 2.02	0.25 ± 0.03	0.28 ± 0.03	1.66 ± 2.12	1.15 ± 2.02	0.24 ± 0.03	1.71 ± 2.02	0.69 ± 0.83	0.17 ± 0.04
$\log(\sigma)$	1.81 ± 0.93	1.02 ± 0.56	0.92 ± 0.59	1.59 ± 0.70	0.98 ± 0.43	0.80 ± 0.29	1.76 ± 0.93	1.30 ± 0.62	1.05 ± 0.73

6 CONCLUSION

Bayesian inference on partially observed SRN plays a critical role for multi-scale bioprocess mechanism learning. To tackle the critical challenges of biomanufacturing processes, including high complexity, high inherent stochasticity, and very limited and sparse observations on partially observed state with measurement errors, we propose an interpretable Bayesian updating LNA metamodel to approximate the likelihood of heterogeneous online and offline measures, accounting for the structure information of the enzymatic SRN mechanistic model. Then, we develop a MALA sampling algorithm that utilizes the information from the derived likelihood and more efficiently generates posterior samples. The empirical study shows that our proposed LNA assisted Bayesian inference approach has a promising performance, demonstrating its potential to benefit bioprocess mechanisms online learning and digital twin development.

REFERENCES

Anderson, D. F. and T. G. Kurtz. 2011. "Continuous Time Markov Chain Models for Chemical Reaction Networks". In *Design and Analysis of Biomolecular Circuits: Engineering Approaches to Systems and Synthetic Biology*, edited by H. Koeppl, G. Setti, M. di Bernardo, and D. Densmore, 3–42. New York, NY: Springer New York.

Archambeau, C., D. Cornford, M. Opper, and J. Shawe-Taylor. 2007. "Gaussian Process Approximations of Stochastic Differential Equations". In *Gaussian Processes in Practice*, edited by N. D. Lawrence, A. Schwaighofer, and J. Quiñonero Candela, 1–16. Bletchley Park, UK: Proceedings of Machine Learning Research.

Chewi, S., C. Lu, K. Ahn, X. Cheng, T. Le Gouic and P. Rigollet. 2021. "Optimal Dimension Dependence of the Metropolis-Adjusted Langevin Algorithm". In *Proceedings of Thirty Fourth Conference on Learning Theory*, edited by M. Belkin and S. Kpotufe, 1260–1300. Boulder, Colorado, USA: Proceedings of Machine Learning Research.

Fearnhead, P., V. Giagos, and C. Sherlock. 2014. "Inference for Reaction Networks Using the Linear Noise Approximation". *Biometrics* 70(2):457–466.

Ferm, L., P. Lötstedt, and A. Hellander. 2008. "A Hierarchy of Approximations of the Master Equation Scaled by a Size Parameter". *Journal of Scientific Computing* 34(2):127–151.

Garcia, C. A., A. Otero, P. Felix, J. Presedo and D. G. Marquez. 2017. "Nonparametric Estimation of Stochastic Differential Equations with Sparse Gaussian Processes". *Physical Review E* 96(2):022104.

Gillespie, D. T. 1977. "Exact Stochastic Simulation of Coupled Chemical Reactions". *The Journal of Physical Chemistry* 81(25):2340–2361.

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

Hillson, N., M. Caddick, Y. Cai, J. A. Carrasco, M. W. Chang, N. C. Curach *et al*. 2019. "Building a Global Alliance of Biofoundries". *Nature Communications* 10(1):2040.

Kloeden, P. and E. Platen. 1992. *Numerical Solution of Stochastic Differential Equations*. 1st ed. Heidelberg: Springer Berlin.

Prague, M., D. Commenges, J. Guedj, J. Drylewicz and R. ThiéBaut. 2013. "NIMROD: A Program for Inference via a Normal Approximation of the Posterior in Models with Random Effects based on Ordinary Differential Equations". *Computer Methods and Programs in Biomedicine* 111(2):447–458.

Rao, C. V. and A. P. Arkin. 2003. "Stochastic Chemical Kinetics and the Quasi-Steady-State Assumption: Application to the Gillespie Algorithm". *The Journal of Chemical Physics* 118(11):4999–5010.

Ruttor, A. and M. Opper. 2009. "Efficient Statistical Inference for Stochastic Reaction Processes". *Physical Review Letters* 103(23):230601.

Xie, W., K. Wang, H. Zheng, and B. Feng. 2022. "Sequential Importance Sampling for Hybrid Model Bayesian Inference to Support Bioprocess Mechanism Learning and Robust Control". In *2022 Winter Simulation Conference (WSC)*, 2282–2293 https://doi.org/10.1109/WSC57314.2022.10015302.

Yang, S., S. W. Wong, and S. Kou. 2021. "Inference of Dynamic Systems from Noisy and Sparse Data via Manifold-Constrained Gaussian Processes". *Proceedings of the National Academy of Sciences* 118(15):e2020397118.

AUTHOR BIOGRAPHIES

WANDI XU is Ph.D. candidate in Mechanical and Industrial Engineering (MIE) at Northeastern University. Her research interests include machine learning, model inference, and computer simulation. Her email address is xu.wand@northeastern.edu.

WEI XIE is an assistant professor in MIE at Northeastern University. Her research interests include interpretable AI/ML, computer simulation, model uncertainty, and stochastic optimization. Her email address is w.xie@northeastern.edu.