Neural Networks 193 (2026) 107893

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Full Length Article

A new local time-decoupled squared Wasserstein-2 method for training stochastic neural networks to reconstruct uncertain parameters in dynamical systems

Mingtao Xia [a,*], Qijing Shen [b], Philip K. Maini [c], Eamonn A. Gaffney [c], Alex Mogilner [d]

[a] Department of Mathematics, University of Houston, Philip Guthrie Hoffman Hall, 3551 Cullen Blvd, Houston, TX, 77204, USA
[b] Nuffield Department of Medicine, University of Oxford, Henry Wellcome Building for Molecular Physiology, Old Road, Oxford, OX3 7BN, United Kingdom
[c] Mathematical Institute, University of Oxford, Radcliffe Observatory, Andrew Wiles Building, Oxford, OX2 6GG, United Kingdom
[d] Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY, 10012, USA

ARTICLE INFO

Keywords:
Uncertainty quantification
Stochastic neural network
Wasserstein distance
Dynamical systems

ABSTRACT

In this work, we propose and analyze a new local time-decoupled squared Wasserstein-2 method for reconstructing the distribution of unknown parameters in dynamical systems from a finite number of observed temporal trajectories. Specifically, we show that a stochastic neural network model, which can be effectively trained by minimizing our proposed local time-decoupled squared Wasserstein-2 loss function, is an effective model for approximating the distribution of uncertain model parameters in dynamical systems. Through several numerical examples, we showcase the effectiveness of our proposed method in reconstructing the distribution of parameters in different dynamical systems.

1. Introduction

The inverse problem of reconstructing a noisy dynamical system from time-series data finds wide applications across different disciplines. Efficient algorithms to solve such inverse-type problems advance different fields including inferring neural circuit dynamics from spiking data (Pillow et al., 2008) in neuroscience, modeling and predicting complex weather patterns from historical data (Carrassi et al., 2018) in climate science, uncovering disease transmission dynamics from infection case counts over time (Roda et al., 2020) in epidemiology, and deducing reaction rates from experimental concentration-time profiles in reaction kinetics in biochemistry (Loskot et al., 2019). However, such inverse-type problems pose substantial mathematical and computational challenges, particularly when data are limited and noisy, motivating ongoing research into novel algorithms and theoretical frameworks to improve model reconstruction accuracy and efficiency.

In this paper, we study the inverse problem of inferring the distribution of model parameters for several dynamical systems including ordinary differential equations (ODEs), partial differential equations (PDEs), and stochastic differential equations (SDEs) from time-series data or spatiotemporal data. Existing methods for such problems can be broadly categorized into traditional statistical approaches and modern data-driven techniques. Traditional statistical methods often involve parameter estimation frameworks. For example, linear and nonlinear regression methods play a role in simpler systems where the functional form of the model is partially known (Chib & Greenberg, 2002). Furthermore, maximum likelihood estimation and Bayesian inference methods (Cresswell, 2015; Marin et al., 2021) are often adopted. Maximum likelihood estimation optimizes the likelihood of model parameter values in a proposed model from observed data, while Bayesian methods incorporate prior information and compute posterior distributions. These approaches are widely used in applications such as reaction network reconstruction and epidemiological modeling. On the other hand, data-driven methods, leveraging advances in machine learning, offer a complementary toolkit for inverse problems. For example, neural networks and reservoir computing frameworks have been successful in reconstructing chaotic systems and inferring governing equations directly from data (Brunton et al., 2016; Pathak et al., 2018). Sparse identification of nonlinear dynamics (SINDy) has emerged as a powerful tool for discovering interpretable dynamical systems by identifying a parsimonious set of governing equations from time-series data (Brunton et al., 2016). Gaussian process regression has proven effective for nonparametric inference, especially in uncertainty quantification (Raissi et al., 2017). Hybrid approaches, which integrate data-driven and traditional statistical methods such as the physics-informed neural network method (Raissi et al., 2019), are also gathering increasing attention as they take

* Corresponding author.
E-mail address: mxia4@uh.edu (M. Xia).

https://doi.org/10.1016/j.neunet.2025.107893
Received 6 March 2025; Received in revised form 9 July 2025; Accepted 20 July 2025
Available online 5 August 2025

advantage of both statistical methods and data-driven methods and provide more interpretable machine-learning tools.

The Wasserstein distance, which can effectively measure the discrepancy between two probability distributions (Balasubramanian et al., 2024; Panaretos & Zemel, 2019; Villani , 2009), was utilized for various uncertainty quantification (UQ) tasks. In (Bernton et al., 2019), the Wasserstein distance was proposed to estimate model parameters that govern a probabilistic model. Compared to empirical maximum likelihood estimates, using the Wasserstein distance to estimate model parameters in uncertainty models can be more efficient and accurate (Blanchet & Kang, 2021). However, previous methods mainly focused on point estimates of model parameters (inferring the exact values of the parameters) and could not quantify the uncertainty in the model parameters, i.e., they cannot reconstruct a distribution of the model parameters from data. Recently, a time-decoupled Wasserstein-2 (W_2) distance method, which compares the distributions of two stochastic processes across different time points, has been revealed to be an efficient loss function for reconstructing intrinsically noisy stochastic processes such as pure-diffusion processes and jump-diffusion processes using parameterized neural networks (Xia et al., 2024a,b). Furthermore, in Xia and Shen (2024), a local squared W_2 method, which adopts a "neighborhood technique" to enlarge the amount of available data, was proposed to infer the distribution of y given observed data x in the uncertainty model $y = f(x, \omega)$ where ω are uncertain latent unobserved variables. However, for the specific problem of inferring unknown parameters in deterministic or stochastic dynamical models, these two methods are not suitable: the time-decoupled W_2 distance method assumes that there is no uncertainty in the underlying model (i.e., the ODE or SDE to be reconstructed is fixed), and the local squared W_2 method is not directly applicable to parameter inference problems. Two major difficulties arise: first, an efficient model is required to approximate the distribution of parameters in a dynamical system; second, it is necessary to distinguish uncertainty in model parameters from uncertainty in the initial state of the dynamical systems and intrinsic stochasticity in the dynamical system, such as the Wiener process in SDEs.

In this work, we propose and analyze a local time-decoupled squared W_2 method, which builds upon the time-decoupled squared W_2 method (Xia et al., 2024b) for efficiently reconstructing the underlying dynamics from noisy time-series data and the local squared W_2 method (Xia & Shen, 2024) for handling the uncertainty in the initial state. This in turn is implemented to infer the distribution of model parameters in deterministic or stochastic dynamical systems given a finite number of observations. As an illustration, consider the following ODE:

$$\mathrm{d}\boldsymbol{X}(t;\theta) = \boldsymbol{f}(\boldsymbol{X}(t;\theta), t; \theta)\mathrm{d}t,\ \boldsymbol{X} \in \mathbb{R}^d,\ \in [0,T],\ \boldsymbol{X}(0) \sim \nu_0,\ \theta \sim \mu, \quad (1.1)$$

where $\theta \in \mathbb{R}^\ell$ is a continuous random variable representing uncertain parameters in the ODE and $\boldsymbol{f} : \mathbb{R}^{d+1+\ell} \to \mathbb{R}^d$. ν_0 is a probability measure defined on the Borel σ-algebra $\mathcal{B}(\mathbb{R}^d)$. For each realization of the ODE (1.1), θ is sampled independently. Thus, we obtain different trajectories by solving the ODE (1.1) multiple times due to uncertainty in model parameters. We use another ODE model as an approximation to Eq. (1.1):

$$\mathrm{d}\hat{\boldsymbol{X}}(t;\hat{\theta}) = \boldsymbol{f}(\hat{\boldsymbol{X}}(t;\hat{\theta}), t; \hat{\theta})\mathrm{d}t,\ \hat{\boldsymbol{X}} \in \mathbb{R}^d,\ t \in [0,T],\ \hat{\boldsymbol{X}}(0) = \boldsymbol{X}_0,\ \hat{\theta} \sim \hat{\mu}. \quad (1.2)$$

Here, $\hat{\theta} \in \mathbb{R}^\ell$ is another continuous random variable and denotes uncertain parameters in the approximate ODE. We aim to construct the probability density function $\hat{\mu}$, i.e. the distribution of $\hat{\theta} \in \mathbb{R}^\ell$, such that $\hat{\mu}$ can match μ well in Eq. (1.1). Specifically, we show that minimizing our proposed loss function can lead to efficient training of a stochastic neural network (SNN) model with weight uncertainty.

SNNs are effective in uncertainty quantification, generative modeling, and time-series analysis (Gal & Ghahramani, 2016; Rezende et al., 2014; Senan et al., 2017; Vadivel et al., 2020). Instead of deterministic outputs, SNNs introduce randomness in the weights and/or biases (Blundell et al., 2015; Yu et al., 2021) or utilize stochastic neurons whose outputs are binary (Tang & Salakhutdinov, 2013). Theoretical results exist for the approximation errors for certain types of SNNs, such as the approximation error of a single-layer SNN (Gonon et al., 2023) as well as the universal approximation ability of dropout neural networks (Manita et al., 2022). In this work, we prove that the SNN model we use can serve as an effective model to approximate a general random field model in the W_2 metric under moderate assumptions. Our result generalizes the universal approximation ability property of deterministic multilayer feedforward neural networks for approximating deterministic functions (Hornik et al., 1989; Leshno et al., 1993) to SNNs for approximating random fields. As a special case, when the input of the SNN is fixed, we show that the output of the SNN can approximate any continuous random variable under moderate assumptions, making it an ideal surrogate model for reconstructing the distribution of unknown parameters in dynamical systems. Compared with traditional Bayesian methods, our proposed approach has the advantage of not requiring any knowledge or prior distributions of model parameters and directly reconstructing model parameter distributions from time-series data. Compared to other data-driven methods, such as Bayesian neural networks (Neal, 2012), generative modeling methods (Böhm et al., 2019), and Wasserstein generative adversarial networks (WGANs) that train a generator and a discriminator (Arjovsky et al., 2017; Boukraichi et al., 2022), our method is more physics-informed and provides more insights and interpretability of the dynamical system. Our method directly outputs the distribution of unknown parameters governing the dynamical system. Furthermore, our method does not require deep neural networks and we shall show that shallow SNNs with hundreds of neurons are capable of reconstructing the joint distribution of several model parameters when inputting only a few hundred trajectories as training data.

The main contributions of our work are as follows:

- We propose and analyze a local time-decoupled squared W_2 method for the direct reconstruction of model parameters in specific dynamical systems including ODEs, SDEs, and PDEs from time-series or spatiotemporal data. Our method takes into account both uncertainties in the initial state as well as intrinsic fluctuations, *e.g.* Wiener processes, of the dynamical system when reconstructing model parameters.
- We analyze an SNN model whose weights are sampled from independent normal distributions. We prove that this SNN model can approximate any multidimensional continuous random variable under moderate assumptions. Furthermore, this SNN model can be trained by direct minimization of our local time-decoupled squared W_2 loss function.
- Through numerical experiments, we showcase the effectiveness of our proposed method for reconstructing the distribution of model parameters in several deterministic and stochastic dynamical systems.

The structure of this paper is as follows. In Section 2, we analyze a local time-decoupled squared W_2 loss function for reconstructing model parameters in dynamical systems. In Section 3, we prove that an SNN model can approximate a continuous random variable in the squared W_2 sense, which makes this SNN model an ideal approximate model for reconstructing the distribution of uncertain parameters in dynamical systems. In Section 4, we carry out numerical experiments and showcase the effectiveness of training the SNN model by minimizing our proposed local function on the reconstruction of uncertain model parameters in different dynamical systems. In Section 5, we summarize our results and propose potential future research directions.

2. A local time-decoupled squared W_2 loss function

In this section, we propose and analyze a local time-decoupled squared W_2 method for reconstructing the distribution of uncertain parameters in specific dynamical systems. Our local time-decoupled squared W_2 method integrates the time-decoupled squared W_2-distance method proposed in Xia et al. (2024a,b) and the local squared W_2-distance method in Xia and Shen (2024). Therefore, both intrinsic

fluctuations in the dynamical systems and uncertainties in the initial condition can be taken into account. First, we introduce the W_2 distance between distributions associated with two multidimensional random variables.

Definition 2.1. *For two d-dimensional random variables*

$$\boldsymbol{X} = (X_1, \ldots, X_d),\ \hat{\boldsymbol{X}} = (\hat{X}_1, \ldots, \hat{X}_d) \in \mathbb{R}^d \tag{2.1}$$

with associated probability measures $\nu, \hat{\nu}$, respectively, the W_2-distance $W_2(\nu, \hat{\nu})$ between their probability measures is defined as

$$W_2(\nu, \hat{\nu}) := \inf_{\pi(\nu,\hat{\nu})} \mathbb{E}_{(\boldsymbol{X},\hat{\boldsymbol{X}})\sim\pi(\nu,\hat{\nu})}\left[\|\boldsymbol{X} - \hat{\boldsymbol{X}}\|^2\right]^{\frac{1}{2}}. \tag{2.2}$$

In Eq. (2.2) and throughout this paper, the norm $\|\cdot\|$ denotes the l^2 norm of a vector: $\|\boldsymbol{X}\| := \left(\sum_{i=1}^d (X_i)^2\right)^{\frac{1}{2}}$. $\pi(\nu, \hat{\nu})$ is a coupling probability measure which iterates over all coupled distributions of $\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t)$, defined by the condition:

$$\begin{cases} \boldsymbol{P}_{\pi(\nu,\hat{\nu})}\left(A \times \mathbb{R}^d\right) = \boldsymbol{P}_{\nu}(A), \\ \boldsymbol{P}_{\pi(\nu,\hat{\nu})}\left(\mathbb{R}^d \times A\right) = \boldsymbol{P}_{\hat{\nu}}(A), \end{cases} \quad \forall A \in \mathcal{B}(\mathbb{R}^d), \tag{2.3}$$

where $\mathcal{B}(\mathbb{R}^d)$ denotes the Borel σ-algebra associated with the space of d-dimensional functions in $\mathbb{R}^d$.

Next, we define the local squared W_2 distance for the probability measures associated with the trajectories of two dynamical systems, which builds upon the local squared W_2 distance introduced in Xia and Shen (2024).

Definition 2.2. *The **local squared W_2 distance** between the probability measures associated with two dynamical systems $\{\boldsymbol{X}(t)\}_{t\in[0,T]}, \{\hat{\boldsymbol{X}}(t)\}_{t\in[0,T]}$ at a specific time t is defined by:*

$$W_{2,\delta}^{2,e}\left(\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t)\right) := \int_{\mathbb{R}^d} W_2^2\left(\nu^e_{\boldsymbol{X}_0,\delta}(t), \hat{\nu}^e_{\boldsymbol{X}_0,\delta}(t)\right)\nu_0^e(\mathrm{d}\boldsymbol{X}_0). \tag{2.4}$$

Here, $\nu_0^e(\cdot)$ is the empirical distribution of the initial condition for $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$. $\nu^e_{\boldsymbol{X}_0,\delta}(t)$ and $\hat{\nu}^e_{\boldsymbol{X}_0,\delta}(t)$ are the empirical conditional probability distributions of $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ at time t conditioned on $\|\boldsymbol{X}(0) - \boldsymbol{X}_0\| \le \delta$ and $\|\hat{\boldsymbol{X}}(0) - \boldsymbol{X}_0\| \le \delta$ for a given initial state $\boldsymbol{X}_0$, respectively.

Now, we define the **local time-decoupled squared W_2** distance between the probability measures associated with trajectories of two dynamical systems.

Definition 2.3. *Let $0 = t_0 < t_1 < \ldots < t_n = T$ be a time discretization mesh in $[0, T]$. The **local time-decoupled squared W_2 distance** between the distributions associated with two dynamical systems $\{\boldsymbol{X}\}_{t\in[0,T]}, \{\hat{\boldsymbol{X}}\}_{t\in[0,T]}$ at time t is defined by:*

$$\begin{aligned} \tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X}, \hat{\boldsymbol{X}}) &:= \int_0^T W_{2,\delta}^{2,e}(\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t))\mathrm{d}t \\ &= \lim_{n\to\infty, \max(t_{i+1}-t_i)\to 0} \sum_{i=0}^{n-1} W_{2,\delta}^{2,e}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))(t_{i+1} - t_i), \end{aligned} \tag{2.5}$$

where $W_{2,\delta}^{2,e}(\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t))$ is the local squared W_2 distance in Definition 2.2. Empirically, we use

$$\tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X}, \hat{\boldsymbol{X}}) \approx \sum_{i=0}^{n-1} W_{2,\delta}^{2,e}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))(t_{i+1} - t_i), \tag{2.6}$$

as an approximation to Eq. (2.5) for numerically calculating the local time-decoupled squared W_2 distance loss function. Under some technical conditions, the approximation error of using the time-discretized RHS of Eq. (2.6) to approximate the local time-decoupled squared W_2 distance loss function in Eq. (2.5) is $O(\max_{i=1}^{n-1}(t_{i+1} - t_i))$ for ODEs and $O\left(\sqrt{\max_{i=0}^{n-1}(t_{i+1} - t_i)}\right)$ for jump-diffusion processes, which will be analyzed in the proof of Theorem 2.1 for ODEs and in the proof of Corollary 2.1 for jump-diffusion processes, respectively.

The loss function Eq. (2.6) was also used to reconstruct the dynamics of an ODE using a parameterized neural network in Xia and Shen (2024). Yet, there is no understanding of why minimizing the loss function Eq. (2.6) leads to the successful reconstruction of the *distribution of parameters* underlying a dynamical system. Additionally, intrinsic stochasticity, such as Wiener processes in stochastic dynamical systems, was not considered in Xia and Shen (2024). In this section, we shall show that: i) the local time-decoupled squared W_2 distance in Eq. (2.5) is well-defined in several deterministic or stochastic dynamical systems and ii) minimizing the loss function in Eq. (2.6) is a necessary condition for the reconstruction of the distribution of parameters in dynamical systems.

First, we prove that the local time-decoupled squared W_2 distance is well-defined in some typical dynamical systems including ODEs and certain SDEs. We can prove that the local squared W_2 distance between the probability measures associated with $\{\boldsymbol{X}\}_{t\in[0,T]}$ and $\{\hat{\boldsymbol{X}}\}_{t\in[0,T]}$, which are trajectories generated by solving the two ODEs (1.1) and (1.2), is well-defined. In Eqs. (1.1), (1.2), and the models we study below, we assume that the model parameters are independent of the initial condition, *e.g.*, in Eqs. (1.1), (1.2) θ is independent of $\boldsymbol{X}(0)$ and $\hat{\theta}$ is independent of $\hat{\boldsymbol{X}}(0)$.

Theorem 2.1. *Suppose*

$$\sup_{\boldsymbol{X}(0),\theta,t} \|\boldsymbol{X}(t)\|^2 \le X, \quad \sup_{\boldsymbol{X}(0),\hat{\theta},t} \|\hat{\boldsymbol{X}}(t)\|^2 \le \hat{X} \tag{2.7}$$

are uniformly bounded, where $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ are solutions to the ODEs (1.1) and (1.2), respectively. Furthermore, we assume that $\boldsymbol{f}$ is continuous and uniformly bounded. Then, the limit

$$\lim_{\max(t_{i+1}-t_i)\to 0} \sum_{i=0}^{n-1} W_{2,\delta}^{2,e}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))(t_{i+1} - t_i) \tag{2.8}$$

on the RHS of Eq. (2.5) exists.

We provide a proof of Theorem 2.1 in Appendix A, which is similar to the proof of Theorem 3.1 in Xia et al. (2024a). Theorem 2.1 can be extended to reveal that the local time-decoupled squared W_2 distance between probability measures associated with two noisy jump-diffusion processes is also well-defined. We can prove the following corollary.

Corollary 2.1. *Consider the following two d-dimensional jump-diffusion processes:*

$$\begin{aligned} \mathrm{d}\boldsymbol{X}(t) &= \boldsymbol{f}(\boldsymbol{X}(t), t; \theta)\mathrm{d}t + \boldsymbol{\sigma}(\boldsymbol{X}(t), t; \theta)\mathrm{d}\boldsymbol{B}_t \\ &\quad + \int_U \boldsymbol{\beta}(\boldsymbol{X}(t), \xi, t; \theta)\tilde{N}(\mathrm{d}t, \gamma(\mathrm{d}\xi)),\ \boldsymbol{X}(0) \sim \nu_0 \end{aligned} \tag{2.9}$$

and

$$\begin{aligned} \mathrm{d}\hat{\boldsymbol{X}}(t) &= \boldsymbol{f}(\hat{\boldsymbol{X}}(t), t; \hat{\theta})\mathrm{d}t + \boldsymbol{\sigma}(\hat{\boldsymbol{X}}(t), t; \hat{\theta})\mathrm{d}\hat{\boldsymbol{B}}_t \\ &\quad + \int_U \boldsymbol{\beta}(\hat{\boldsymbol{X}}(t), \xi, t; \hat{\theta})\hat{N}\left(\mathrm{d}t, \gamma(\mathrm{d}\xi)\right),\ \hat{\boldsymbol{X}}(0) = \boldsymbol{X}(0). \end{aligned} \tag{2.10}$$

In Eq. (2.9), $\boldsymbol{f} : \mathbb{R}^{d+\ell+1} \to \mathbb{R}^d$ and $\boldsymbol{\sigma} : \mathbb{R}^{d+\ell+1} \to \mathbb{R}^{d\times m}$ denote the drift and diffusion functions of the SDE, respectively; ν_0 is a probability measure defined on the Borel σ-algebra $\mathcal{B}(\mathbb{R}^d)$; $\boldsymbol{B}(t)$ represents an m-dimensional standard Brownian motion; $\tilde{N}(\mathrm{d}t, \gamma(\mathrm{d}\xi))$ is a compensated Poisson process independent of $\boldsymbol{B}_t$ defined as follows:

$$\tilde{N}(\mathrm{d}t, \gamma(\mathrm{d}\xi)) := N(\mathrm{d}t, \gamma(\mathrm{d}\xi)) - \gamma(\mathrm{d}\xi)\mathrm{d}t, \tag{2.11}$$

where $N(\mathrm{d}t, \gamma(\mathrm{d}\xi))$ is a Poisson process with intensity $\gamma(\mathrm{d}\xi)\mathrm{d}t$, and $\gamma(\mathrm{d}\xi)$ is a measure defined on $U \subseteq \mathbb{R}$, the measure space of the Poisson process. $\hat{N}(\mathrm{d}t, \gamma(\mathrm{d}\xi))$ is another compensated Poisson process of intensity $\gamma(\mathrm{d}\xi)\mathrm{d}t$ and independent of $\boldsymbol{B}_t, \tilde{N}_t$ in Eq. (2.9) and $\hat{\boldsymbol{B}}_t$ in Eq. (2.10). $\theta, \hat{\theta} \in \mathbb{R}^\ell$ are uncertain model parameters. We assume that $\boldsymbol{f}, \boldsymbol{\sigma}, \boldsymbol{\beta}$ are continuous and uniformly bounded. Then

$$\tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X}, \hat{\boldsymbol{X}}) := \lim_{\max(t_{i+1}-t_i)\to 0} \sum_{i=0}^{n-1} W_{2,\delta}^{2,e}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))(t_{i+1} - t_i) \tag{2.12}$$

exists. Furthermore,

$$\begin{aligned}&\Big|\sum_{i=0}^{n-1} W_{2,\delta}^{2,e}(\boldsymbol{X}(t_i),\hat{\boldsymbol{X}}(t_i))(t_{i+1}-t_i) - \tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X},\hat{\boldsymbol{X}})\Big| \\ &\quad\le 2(X+\hat{X})T\max_i\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right),\end{aligned} \tag{2.13}$$

where $\Delta t := \max_{i=0}^{n-1}(t_{i+1}-t_i)$*, and* $\nu(t_i)$ *and* $\hat{\nu}(t_i)$ *are the probability measures of* $\boldsymbol{X}(t_i)$ *and* $\hat{\boldsymbol{X}}(t_i)$*, respectively. In Eq. (2.13),*

$$\begin{aligned}F_i &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d} f_\ell^2(\boldsymbol{X}(t^-),t^-;\theta)\mathrm{d}t\right],\\ \hat{F}_i &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d} f_\ell^2(\hat{\boldsymbol{X}}(t^-),t^-;\hat{\theta})\mathrm{d}t\right],\\ \Sigma_i &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d}\sum_{j=1}^{m} \sigma_{\ell,j}^2(\boldsymbol{X}(t^-),t^-;\theta)\mathrm{d}t\right],\\ \hat{\Sigma}_i &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d}\sum_{j=1}^{m} \hat{\sigma}_{\ell,j}^2(\hat{\boldsymbol{X}}(t^-),t^-;\theta)\mathrm{d}t\right],\\ B_i &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d}\int_U \beta_\ell^2(\boldsymbol{X}(t^-),\xi,t^-;\theta)\gamma(\mathrm{d}\xi)\mathrm{d}t\right],\\ \hat{B}_i &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d}\int_U \hat{\beta}_\ell^2(\hat{\boldsymbol{X}}(t^-),\xi,t^-;\hat{\theta})\gamma(\mathrm{d}\xi)\mathrm{d}t\right],\\ X &:= \sup_{X_0,\theta,t}\mathbb{E}[\|\boldsymbol{X}(t)\|^2]^{\frac{1}{2}},\quad \hat{X} := \sup_{X_0,\theta,t}\mathbb{E}[\|\hat{\boldsymbol{X}}(t)\|^2]^{\frac{1}{2}},\end{aligned} \tag{2.14}$$

where $f_\ell(\boldsymbol{X}(t^-),t^-;\theta), \sigma_{\ell,j}(\boldsymbol{X}(t^-),t^-;\theta)$*, and* $\beta_\ell(\boldsymbol{X}(t^-),\xi,t^-;\theta)$ *refer to the left-hand limits:*

$$\begin{aligned}f_\ell(\boldsymbol{X}(t^-),t^-;\theta) &= \lim_{s\to t,s<t} f_\ell(\boldsymbol{X}(s),s;\theta),\\ \sigma_{\ell,j}(\boldsymbol{X}(t^-),t^-;\theta) &= \lim_{s\to t,s<t} \sigma_{\ell,j}(\boldsymbol{X}(s),s;\theta),\\ \beta_\ell(\boldsymbol{X}(t^-),t^-,\xi;\theta) &= \lim_{s\to t,s<t} \beta_\ell(\boldsymbol{X}(s),s,\xi;\theta).\end{aligned} \tag{2.15}$$

The proof of Corollary 2.1 is in Appendix B. Next, we show that the local time-decoupled squared W_2 distance $\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}})$ in Definition 2.3 can be bounded by the squared W_2 distance between the two probability measures associated with the two sets of parameters θ and $\hat{\theta}$ in Eqs. (1.1) and (1.2), which implies the necessity of minimizing the local time-decoupled squared W_2 distance $\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}})$ if we wish to match the distribution of θ using the distribution of the reconstructed $\hat{\theta}$.

Theorem 2.2. *Suppose the drift, diffusion, and jump functions in the two jump-diffusion processes Eqs. (2.9) and (2.10) satisfy the following Lipschitz condition:*

$$\begin{aligned}&\sum_{i=1}^{d}|f_i(\boldsymbol{X},t;\theta)-f_i(\hat{\boldsymbol{X}},t;\hat{\theta})| \le C(\|\boldsymbol{X}-\hat{\boldsymbol{X}}\|+\|\theta-\hat{\theta}\|),\\ &\sum_{i=1}^{d}|\sigma_{i,j}(\boldsymbol{X},t;\theta)-\sigma_{i,j}(\hat{\boldsymbol{X}},t;\hat{\theta})| \le C(\|\boldsymbol{X}-\hat{\boldsymbol{X}}\|+\|\theta-\hat{\theta}\|),\ j=1,\dots,m,\\ &\sum_{i=1}^{d}|\beta_i(\boldsymbol{X},\xi,t;\theta)-\beta_i(\hat{\boldsymbol{X}},\xi,t;\hat{\theta})| \le C(\|\boldsymbol{X}-\hat{\boldsymbol{X}}\|+\|\theta-\hat{\theta}\|),\\ &\quad\forall\,\boldsymbol{X},\hat{\boldsymbol{X}}\in\mathbb{R}^d,\ \forall\theta,\hat{\theta}\in\mathbb{R}^\ell,\ C<\infty.\end{aligned} \tag{2.16}$$

In Eq. (2.16), f_i *and* β_i *denote the* i*th component of* $\boldsymbol{f}$ *and* $\boldsymbol{\beta}$*, respectively, and* $\sigma_{i,j}$ *denotes the* (i,j) *element of the matrix* $\boldsymbol{\sigma}$ *in Eq. (2.9). Furthermore, we assume that the assumptions in Corollary 2.1 hold and the sixth-order moments*

$$\mathbb{E}[\|\theta\|^6]\le\Theta_6,\ \mathbb{E}[\|\hat{\theta}\|^6]\le\hat{\Theta}_6 \tag{2.17}$$

are uniformly bounded. Then, $\tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X},\hat{\boldsymbol{X}})$ *can be bounded by the squared* W_2 *distance* $W_2^2(\mu,\hat{\mu})$*:*

$$\begin{aligned}\mathbb{E}\left[\tilde{W}_{2,\delta}^{2,e}(\boldsymbol{X},\hat{\boldsymbol{X}})\right] &\le 8C_0T\delta^2\exp(C_0T)+\tfrac{6C_1}{C_0}T\exp(C_0T)\\ &\quad\times\left(W_2^2(\mu,\hat{\mu})+2C_3\mathbb{E}\left[h(N^{\#}(\boldsymbol{X}_0;\delta),\ell)\cdot\left(\Theta_6^{\frac{1}{3}}+\hat{\Theta}_6^{\frac{1}{3}}\right)\right]\right),\end{aligned} \tag{2.18}$$

where μ *and* $\hat{\mu}$ *are the probability measures associated with* θ *and* $\hat{\theta}$*, respectively. In Eq. (2.18),* C_0, C_1, C_2 *are three constants, and*

$$h(N,\ell) := \begin{cases}2N^{-\frac{1}{2}}(\log(1+N)+1), & \ell\le 4,\\ 2N^{-\frac{2}{\ell}}, & \ell>4.\end{cases} \tag{2.19}$$

In Eq. (2.18), $N^{\#}(\boldsymbol{X}_0;\delta)$ *refers to the number of trajectories in the data set such that their initial conditions satisfy* $\|\boldsymbol{X}(0)-\boldsymbol{X}_0\|\le\delta$*.*

We prove Theorem 2.2 in Appendix C. Theorem 2.2 implies that minimizing the local time-decoupled squared W_2 distance $\tilde{W}_{2,\delta}^{2,\mathrm{e}}$ is a necessary condition if we wish to match the distribution of θ with the distribution of $\hat{\theta}$. In Eq. (2.18), there is a trade-off between the first term and the third term on the RHS: if we increase δ, then the first term on the RHS of Eq. (2.18) will increase but the factor $h(N^{\#}(\boldsymbol{X}_0;\delta);\ell)$ in the last term will decrease. In (Xia & Shen, 2024), it is found that a δ that is too large may lead to systematic errors, while a δ that is too small is insufficient to quantify the uncertainty. The optimal choice of δ is problem-specific and also depends on the amount of available training data, which may require some fine-tuning. When an appropriate δ is chosen, both the first and last terms on the RHS of Eq. (2.18) can be kept small so that the expected local time-decoupled squared W_2 distance $\mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}})]$ can be well controlled by $W_2^2(\mu,\hat{\mu})$. In this case, minimizing $\mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}})]$ is necessary to minimize $W_2^2(\mu,\hat{\mu})$, i.e., match the distribution of θ with the distribution of $\hat{\theta}$. On the other hand, in Eq. (2.19), the rate of convergence when using the empirical probability measure is $2N^{-\frac{1}{2}}(\log(1+N)+1), \ell\le 4$ or $N^{-\frac{2}{\ell}}, \ell>4$ as the number of observed trajectories increases, which depends only on the dimensionality of the parameter θ instead of on the dimensionality of $\boldsymbol{X}(t)$. Therefore, it is harder to reconstruct the joint distribution of higher-dimensional uncertain parameters due to a slower convergence rate when only finite training trajectories are available.

ODEs can be regarded as special jump-diffusion processes whose diffusion function and jump function are both 0. Thus, Theorem 2.2 can be applied to bound the time-decoupled local squared W_2 distance between the distributions of trajectories of the two ODEs Eqs. (1.1) and (1.2) using the squared W_2 distance between the probability distributions of θ and $\hat{\theta}$. Finally, Theorem 2.2 can also be generalized to the cases of reconstructing parameters in spatiotemporal stochastic partial differential equations (SPDEs). We provide an example of bounding the local time-decoupled squared W_2 distance between solutions to a parabolic SPDE associated with two different sets of model parameters in Appendix D.

Finally, one can also take into account time discretization errors using a time discretization scheme such as the Runge-Kutta scheme for solving ODEs and the strong Itô-Taylor approximation (Kloeden & Platen, 1992) for numerically solving SDEs. Specifically, for stiff problems, a detailed analysis of the numerical implementation is worth carrying out. Additionally, it is also possible to extend Theorem 2.2 to more complicated spatiotemporal dynamics by replacing B_t with more complicated spatiotemporal cylindrical Brownian noise (Liu & Röckner, 2015) or considering spatiotemporal integrodifferential equations (Deng et al., 2025). Such discussions require a more detailed analysis of SPDEs and are thus beyond the scope of this paper.

3. An SNN model for approximating the distribution of continuous random variables

In this section, we analyze an SNN model used in Xia and Shen (2024). We apply this SNN model to reconstruct the distribution of unknown parameters for general dynamical systems. Specifically, we will prove that this SNN model can approximate the probability distribution of any continuous multidimensional random variable under certain technical assumptions in the W_2 distance sense, and the training of this SNN can be done by direct minimization of the local time-decoupled squared W_2 loss function Eq. (2.6) analyzed in Section 2. We sketch the structure of the SNN with weight uncertainty in Fig. 1. All weights in the neural networks $\{w_{i,j,k}\}$ are sampled from independent normal distributions with $w_{i,j,k} \sim \mathcal{N}(a_{i,j,k}, \sigma^2_{i,j,k})$. The biases $\{b_{i,k}\}$ for all i,k are deterministic. The means and variances of the weights $\{a_{i,j,k}\}, \{\sigma^2_{i,j,k}\}$ and the biases $\{b_{i,k}\}$ are optimized through training.

First, we prove that the SNN model in Fig. 1 can approximate any continuous random variable whose probability distribution follows a parameterized multivariate normal distribution in the squared W_2 distance sense. In the following, $\mathcal{N}(\boldsymbol{y}-\boldsymbol{b}, \Sigma)$ denotes the probability density function of a d'-dimensional multivariate normal distribution with mean $\boldsymbol{b}$ and the covariance matrix Σ and takes the form:

$$\mathcal{N}(\boldsymbol{y}-\boldsymbol{b}, \Sigma) := \frac{1}{\sqrt{2\pi}^{d'}} \cdot \frac{1}{|\Sigma|^{\frac{1}{2}}} \cdot \exp\Big(-\tfrac{1}{2}(\boldsymbol{y}-\boldsymbol{b})^T \Sigma (\boldsymbol{y}-\boldsymbol{b})\Big). \tag{3.1}$$

Theorem 3.1. *Let $\boldsymbol{y} \in \mathbb{R}^{d'}$ be a continuous random variable whose probability density is $f_{\boldsymbol{x}}(\boldsymbol{y}), \boldsymbol{x} \in D \subseteq \mathbb{R}^d$ such that $f_{\boldsymbol{x}}(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}-\boldsymbol{b}(\boldsymbol{x}), A(\boldsymbol{x})^T A(\boldsymbol{x}))$. D is a bounded set and $\boldsymbol{x}$ has a probability measure $\gamma(\cdot)$ on D. We make the following assumptions:*

1. *For any sequence of sets $\{D_i\}_{i=1}^{\infty}$, if $D_i \to D$ as $i \to \infty$, then $\lim_{i\to\infty} \gamma(D_i) = \gamma(D) = 1$. Furthermore, for any $\Delta x > 0$, we can find a set of equidistance grids $\{\boldsymbol{x}_i\}_{i=1}^{K} \subseteq D$ such that the distance between two adjacent grids is Δx, $D \subseteq \cup_{i=1}^{K} \otimes_{j=1}^{d} [x_i^j, x_i^j + \Delta x)$, and $\otimes_{j=1}^{d}[x_{i_1}^j, x_{i_1}^j + \Delta x) \cap \otimes_{j=1}^{d}[x_{i_2}^j, x_{i_2}^j + \Delta x) = \emptyset$ if $i_1 \neq i_2$.*
2. *$f_{\boldsymbol{x}}(\boldsymbol{y})$ is uniformly continuous in $\boldsymbol{x}$ such that for any $\epsilon > 0$, there exists $\Delta x > 0$ satisfying:*

$$W_2^2(f_{\boldsymbol{x}}, f_{\tilde{\boldsymbol{x}}}) < \epsilon, \ \forall \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| \le \Delta x, \ \boldsymbol{x}, \tilde{\boldsymbol{x}} \in D. \tag{3.2}$$

3. *$Y := \sup_{\boldsymbol{x}\in D} \|\boldsymbol{b}(\boldsymbol{x})\|^2 + \|A(\boldsymbol{x})^T A(\boldsymbol{x})\|_F^2 < \infty$, where $\|\cdot\|_F$ is the Frobenius norm of a matrix.*

Then, for any $\epsilon_0 > 0$, there exists an SNN model as described in Fig. 1 which uses the ReLU activation and the linear forward propagation such that if we denote the probability density function of the output by $\hat{f}_{\boldsymbol{x}}$ given the input $\boldsymbol{x}$, the following inequality holds:

$$\int_D W_2^2(f_{\boldsymbol{x}}, \hat{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) \le \epsilon_0. \tag{3.3}$$

We prove Theorem 3.1 in Appendix E. Theorem 3.1 can be further generalized and we can prove that the probability density distribution of the output of the SNN model in Fig. 1 can approximate a multivariate Gaussian mixture model (defined in Corollary 3.1 below) in the squared W_2 distance sense. First, we prove the following lemma.

Lemma 3.1. *Let $\boldsymbol{y} \in \mathbb{R}^{d'}$ be a continuous random variable that has the following probability density function of a Gaussian mixture model:*

$$f(\boldsymbol{y}) = \sum_{i=1}^{s} p_i \mathcal{N}\big(\boldsymbol{y}-\boldsymbol{b}_i, A_i^T A_i\big), \ p_i > 0, \ \sum_{i=1}^{s} p_i = 1. \tag{3.4}$$

Suppose another continuous random variable $\hat{\boldsymbol{y}} \in \mathbb{R}^{d'}$ has the following probability density function:

$$\hat{f}(\hat{\boldsymbol{y}}) = \sum_{i=1}^{s} \hat{p}_i \mathcal{N}\big(\hat{\boldsymbol{y}}-\boldsymbol{b}_i, A_i^T A_i\big) + p(\hat{\boldsymbol{y}}), \ \hat{p}_i > 0 \tag{3.5}$$

where $\hat{p}_i \le p_i$, $p(\cdot) : \mathbb{R}^{d'} \to \mathbb{R}^+ \cup \{0\}$ is non-negative, and $\int_{\mathbb{R}^{d'}} \|\hat{\boldsymbol{y}}\|^2 p(\hat{\boldsymbol{y}}) \mathrm{d}\hat{\boldsymbol{y}} \le \infty$. Then, the following bound of the squared W_2 distance between the probability measures of $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ holds:

$$W_2^2(f, \hat{f}) \le 2\Big(\sum_{i=1}^{s} (p_i - \hat{p}_i)\big(\|\boldsymbol{b}_i\|^2 + \|A_i^T A_i\|_F^2\big) + \int_{\mathbb{R}^{d'}} \|\boldsymbol{y}\|^2 p(\boldsymbol{y}) \mathrm{d}\boldsymbol{y}\Big). \tag{3.6}$$

Proof. First, if $p_i = \hat{p}_i, i = 1, \ldots, s$, then $f(\boldsymbol{y}) = \hat{f}(\boldsymbol{y})$ and $W_2^2(f, \hat{f}) = 0$, indicating that Eq. (3.6) holds. Next, we assume that $\sum_{i=1}^{s} \hat{p}_i < 1$. Without loss of generality, we assume that $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ are independent of each other. We define a special coupling probability measure of the random variable $(\boldsymbol{y}, \hat{\boldsymbol{y}}) \in \mathbb{R}^{2d'}$:

$$\begin{aligned}\pi(f, \hat{f})(\boldsymbol{y}, \hat{\boldsymbol{y}}) = &\Big[\sum_{i=1}^{s} \hat{p}_i \mathcal{N}\big(\boldsymbol{y}-\boldsymbol{b}_i, A_i^T A_i\big)\Big]\delta(\boldsymbol{y}-\hat{\boldsymbol{y}}) \\ &+ \frac{1}{p}\Big[\sum_{i=1}^{s} (p_i - \hat{p}_i)\mathcal{N}\big(\boldsymbol{y}-\boldsymbol{b}_i, A_i^T A_i\big)\Big] \cdot p(\hat{\boldsymbol{y}}),\end{aligned} \tag{3.7}$$

where $p := \sum_{i=1}^{s}(p_i - \hat{p}_i) = \int_{\mathbb{R}^{d'}} p(\hat{\boldsymbol{y}}) \mathrm{d}\hat{\boldsymbol{y}}$, and δ is the Dirac delta measure. We can check that the marginal distributions of $\pi(f, \hat{f})$ coincide with $f(\boldsymbol{y})$ and $\hat{f}(\hat{\boldsymbol{y}})$, respectively. Furthermore, we have

$$\begin{aligned} W_2^2(f, \hat{f}) &\le \mathbb{E}_{(\boldsymbol{y},\hat{\boldsymbol{y}})\sim\pi(f,\hat{f})}\Big[\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2\Big] \\ &\le \int_{\mathbb{R}^{2d'}} \frac{1}{p}\Big[\sum_{i=1}^{s}(p_i - \hat{p}_i)\mathcal{N}\big(\boldsymbol{y}-\boldsymbol{b}_i, A_i^T A_i\big)\Big] \cdot p(\hat{\boldsymbol{y}})\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2 \mathrm{d}\boldsymbol{y}\mathrm{d}\hat{\boldsymbol{y}} \\ &\le \int_{\mathbb{R}^{2d'}} \frac{1}{p}\Big[\sum_{i=1}^{s}(p_i - \hat{p}_i)\mathcal{N}\big(\boldsymbol{y}-\boldsymbol{b}_i, A_i^T A_i\big)\Big] \cdot p(\hat{\boldsymbol{y}}) \\ &\quad \cdot 2\big(\|\boldsymbol{y}\|^2 + \|\hat{\boldsymbol{y}}\|^2\big)\mathrm{d}\boldsymbol{y}\mathrm{d}\hat{\boldsymbol{y}} \end{aligned}$$

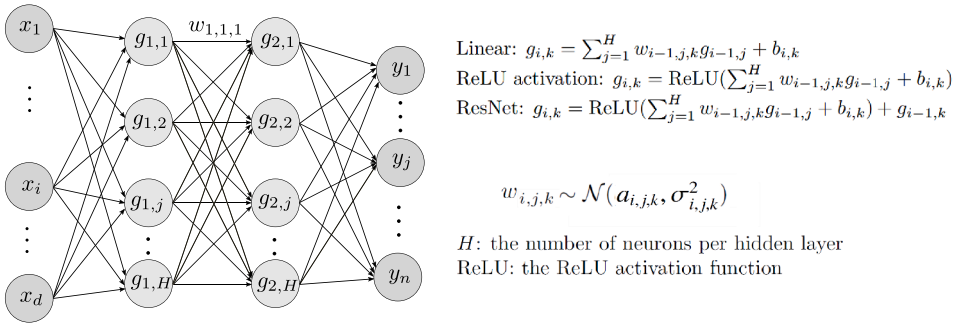

Fig. 1. A sketch of the structure of the neural network model with weight uncertainty used in Xia and Shen (2024) and in this paper. The weights $w_{i,j,k} \sim \mathcal{N}(a_{i,j,k}, \sigma^2_{i,j,k})$ are independently sampled, i.e., w_{i_1,j_1,k_1} is independent of w_{i_2,j_2,k_2} when $(i_1, j_1, k_1) \neq (i_2, j_2, k_2)$. When using this neural network model to make predictions, for each input $\boldsymbol{x} = (x_1, \ldots, x_d) \in D \subseteq \mathbb{R}^d$, we resample all weights $\{w_{i,j,k}\}$ again. For each neuron in the hidden layer, one of the following four forward propagation methods is considered: the linear operation, the ReLU activation, the ELU activation, and the ResNet technique.

$$\leq 2\sum_{i=1}^{s}(p_i-\hat{p}_i)\left(\|\boldsymbol{b}_i\|^2+\|A_i^TA_i\|_F^2\right)+2\int_{\mathbb{R}^{d'}}\|\hat{\boldsymbol{y}}\|^2p(\hat{\boldsymbol{y}})\mathrm{d}\hat{\boldsymbol{y}}, \quad (3.8)$$

which proves the inequality (3.6). □

Next, we show that the SNN model in Fig. 1 can approximate a multivariate Gaussian mixture model in the squared W_2 distance sense.

Corollary 3.1. *Let $\boldsymbol{y}\in\mathbb{R}^{d'}$ be a continuous random variable with a probability density function $f_{\boldsymbol{x}}$, where $\boldsymbol{x}\in D\subseteq\mathbb{R}^d$ is continuous and has a probability density $\gamma(\cdot)$. At each $\boldsymbol{x}$, $f_{\boldsymbol{x}}$ is the probability density function of a Gaussian mixture model:*

$$f_{\boldsymbol{x}}(\boldsymbol{y})=\sum_{r=1}^{s}p_r(\boldsymbol{x})\mathcal{N}(\boldsymbol{y}-\boldsymbol{b}_r(\boldsymbol{x}),A_r^T(\boldsymbol{x})A_r(\boldsymbol{x})),\ \sum_{r=1}^{s}p_r(\boldsymbol{x})=1,\ p_r(\boldsymbol{x})>0. \quad (3.9)$$

We make the following three additional assumptions:

1. *For any sequence of sets $\{D_i\}_{i=1}^{\infty}$, if $D_i\to D$ as $i\to\infty$, then $\lim_{i\to\infty}\gamma(D_i)=\gamma(D)=1$. Additionally, D is a bounded set in $\mathbb{R}^d$ and for any $\Delta x>0$, we can find a set of equidistance grids $\{\boldsymbol{x}_i\}_{i=1}^{K}\subseteq D$ such that $D\subseteq\cup_{i=1}^{K}\otimes_{j=1}^{d}[x_i^j,x_i^j+\Delta\boldsymbol{x})$ and $\otimes_{j=1}^{d}[x_{i_1}^j,x_{i_1}^j+\Delta\boldsymbol{x})\cap\otimes_{j=1}^{d}[x_{i_2}^j,x_{i_2}^j+\Delta\boldsymbol{x})=\emptyset$ if $i_1\neq i_2$.*
2. *$f_{\boldsymbol{x}}(\boldsymbol{y})$ is uniformly continuous in $\boldsymbol{x}$ such that for any $\epsilon>0$, there exists a $\delta>0$:*

$$W_2^2\left(f_{\boldsymbol{x}}(\boldsymbol{y}),f_{\tilde{\boldsymbol{x}}}(\boldsymbol{y})\right)<\epsilon,\ \forall\|\boldsymbol{x}-\tilde{\boldsymbol{x}}\|\leq\delta,\ \forall\boldsymbol{x},\tilde{\boldsymbol{x}}\in D. \quad (3.10)$$

3. *The quantity*

$$\max_{1\leq r\leq s}\left\|A_r^T(\boldsymbol{x})A_r^T(\boldsymbol{x})\right\|_F^2+\|\boldsymbol{b}_r(\boldsymbol{x})\|^2 \quad (3.11)$$

is uniformly bounded for all $\boldsymbol{x}\in D$.

Then, for any positive number $c_0>0$, there exists an SNN with weight uncertainty described in Fig. 1 such that:

$$\int_D W_2^2(f_{\boldsymbol{x}},\hat{f}_{\boldsymbol{x}})\gamma(\mathrm{d}\boldsymbol{x})\leq c_0. \quad (3.12)$$

Here, $\hat{f}_{\boldsymbol{x}}$ is the distribution of the output of the SNN when the input is $\boldsymbol{x}$.

We prove Corollary 3.1 in Appendix F. Finally, we prove that for each continuous random variable $\boldsymbol{y}\in\mathbb{R}^{d'}$ with a probability distribution function $f(\boldsymbol{y})$, under certain conditions, we can find a random variable $\hat{\boldsymbol{y}}\in\mathbb{R}^{d'}$ obeying a Gaussian mixture distribution whose probability distribution function is denoted by $\hat{f}$ such that $\hat{f}$ can approximate f in the W_2 sense. We have the following result.

Theorem 3.2. *Suppose $\boldsymbol{y}=(y_1,\ldots,y_{d'})\in\mathbb{R}^{d'}$ is a continuous random variable with a smooth probability density function $f(\boldsymbol{y})\in L^2(\mathbb{R}^{d'})\cap L^{\infty}(\mathbb{R}^{d'})$. Furthermore, we assume the following conditions hold: 1. $f(\boldsymbol{y})$ is uniformly continuous in $\mathbb{R}^n$ 2.*

$$|f|_{mix}:=\sum_{|\boldsymbol{n}|_0\leq d'}\left\|\partial_{\boldsymbol{n}}^{|\boldsymbol{n}|_0}f\right\|_{L^2}<\infty,\ \left|\sqrt{f}\right|_{mix}<\infty \quad (3.13)$$

where $|\boldsymbol{n}|_0$ is the number of non-zero components in $\boldsymbol{n}$, $\boldsymbol{n}=(n_1,\ldots,n_j)$ satisfying $1\leq n_1<\ldots<n_j\leq d'$, and $\partial_{\boldsymbol{n}}f:=\partial_{y_{n_1}}\ldots\partial_{y_{n_j}}f$.

3. $|fy_i^2|_{mix}<\infty$ and $|fy_i^2y_j^2|_{mix}<\infty$.

Then, for every $\sigma>0,n_0(\sigma)>0$, there exists a probability density function of a Gaussian mixture model:

$$\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y}):=\sum_{i=1}^{(n_0(\sigma)+1)^{d'}}p_i\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i;\sigma^2I_{d'\times d'}),\ \boldsymbol{y}_i\in\mathbb{R}^{d'} \quad (3.14)$$

such that as $\sigma\to0^+$ and $n(\sigma)\to\infty$:

$$\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})\to f(\boldsymbol{y}) \quad (3.15)$$

uniformly in $\mathbb{R}^{d'}$. Furthermore, $\forall\epsilon>0$, there exist $\sigma>0,n_0(\sigma)$ and $\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})$ in Eq. (3.14) such that

$$W_2^2(f,\tilde{f}_{\sigma^2,n_0(\sigma)})\leq24\epsilon. \quad (3.16)$$

In Eq. (3.14), $I_{d'\times d'}$ is a $d'\times d'$ identity matrix.

We prove Theorem 3.2 in Appendix G. For any continuous random variable $\boldsymbol{y}\in\mathbb{R}^{d'}$ with a probability density function f satisfying the assumptions in Theorem 3.2, there exists a random variable $\hat{\boldsymbol{y}}$ whose probability density function is the probability density function of a multivariate Gaussian mixture model denoted by $\tilde{f}_{\sigma^2,n_0}$ such that $W_2^2(f,\tilde{f}_{\sigma^2,n_0})\leq\epsilon$. As a special case of the proof of Corollary 3.1 in Appendix F, there exists an SNN (Fig. 1) which takes the input $n_i^3=x\equiv1$, i.e. this SNN consists of the fourth, fifth, sixth, and seventh layers of the SNN model we constructed in Appendix F, such that $W_2^2(\tilde{f}_{\sigma^2,n_0},\hat{f}_{x\equiv1})\leq\epsilon$, where $\hat{f}_{x\equiv1}$ is the probability density function of the output of the SNN. Thus, we have $W_2^2(f,\hat{f}_{x\equiv1})\leq4\epsilon$, i.e., the SNN model in Fig. 1 can approximate the distribution of any continuous random variable in the squared W_2 sense under the assumptions specified in Theorem 3.2. From Theorem 2.2, minimizing the local time-decoupled squared W_2 loss function Eq. (2.6) is necessary to minimize $W_2^2(\mu,\hat{\mu})$, i.e., to match the distribution of model parameters θ by the distribution of $\hat{\theta}$ in dynamical systems like Eqs. (2.9) and (2.10). Thus, we can train the SNN model in Fig. 1 by direct minimization of the local time-decoupled squared W_2 loss function Eq. (2.6).

Remark: Consider the more general model $\boldsymbol{y}=f(\boldsymbol{x},\theta),\boldsymbol{x}\in\mathbb{R}^d,\boldsymbol{y}\in\mathbb{R}^{d'}$ as in Xia and Shen (2024), where θ is the latent random model parameters (*e.g.* measurement noise). $\boldsymbol{y}$ is a continuous random variable whose distribution is determined by the observed continuous variable $\boldsymbol{x}\in D\subseteq\mathbb{R}^d$. D is a bounded set in $\mathbb{R}^d$, and for any sequence of sets $\{D_i\}_{i=1}^{\infty}$, if $D_i\to D$ as $i\to\infty$, then $\lim_{i\to\infty}\gamma(D_i)=\gamma(D)=1$. Additionally, we assume that for any $\Delta x>0$, we can find a set of equidistance grids $\{\boldsymbol{x}_i\}_{i=1}^{K}\subseteq D$ such that $D\subseteq\cup_{i=1}^{K}\otimes_{j=1}^{d}[x_i^j,x_i^j+\Delta\boldsymbol{x})$ and $\otimes_{j=1}^{d}[x_{i_1}^j,x_{i_1}^j+\Delta\boldsymbol{x})\cap\otimes_{j=1}^{d}[x_{i_2}^j,x_{i_2}^j+\Delta\boldsymbol{x})=\emptyset$ if $i_1\neq i_2$. ω are continuous latent parameters in the model sampled from an unknown distribution. We denote the distribution of $\boldsymbol{y}$ given $\boldsymbol{x}$ by $f_{\boldsymbol{x}}$. Under certain assumptions, for any $\epsilon>0$, there exists an SNN in Fig. 1 such that:

$$\int_D W_2^2(f_{\boldsymbol{x}},\hat{f}_{\boldsymbol{x}})\gamma(\mathrm{d}\boldsymbol{x})\leq\epsilon. \quad (3.17)$$

where $\hat{f}_{\boldsymbol{x}}$ is the probability density function of the output of the SNN given the input $\boldsymbol{x}$. We give a brief discussion on this "universal approximation" property of the SNN model to approximate a family of continuous random variables $\boldsymbol{y}=f(\boldsymbol{x},\theta)$ for all $\boldsymbol{x}\in D\subseteq\mathbb{R}^d$ in Appendix H. Our SNN can approximate a family of probability density functions $f(\boldsymbol{x},\theta),\boldsymbol{x}\in D$ simultaneously, while in Lu and Lu (2020) only a single probability density function $f(\theta)$ dependeing on θ is to be approximated. Furthermore, our SNN utilizes only seven hidden layers and the number of neurons in each layer scales linearly with the dimensionality of either the input or the output variable. This implies that even with a shallow neural network, we might be able to reconstruct a family of uncertainty models $\boldsymbol{y}=f(\boldsymbol{x},\theta)$ characterized by different $\boldsymbol{x}\in D$.

4. Numerical results

In this section, we conduct numerical experiments to test our proposed local squared W_2 method, which involves training the SNN model in Fig. 1 by minimizing the loss function Eq. (J.1), a scaled numerical approximation to the local time-decoupled squared W_2 distance loss function Eq. (2.6). The W_2 distance between two empirical probability measures is then numerically evaluated using the `POT` package of Python in Flamary et al. (2021). A pseudocode of our method is given in Algorithm 1, and a schematic diagram on the training of the SNN using our proposed local time-decoupled squared W_2 loss function Eq. (2.4) in presented in Fig. 2.

Specifically, when applying the SNN model in Fig. 1 for the reconstruction of the distribution for uncertain model parameters, we assume that uncertain model parameters are sampled from the same underlying distribution across all samples and thus always input a scalar 1 as the input into the SNN. Default hyperparameters and training settings are given in Appendix I. We use the ELU activation function to replace the ReLU activation function in Examples 4.1–4.3 to tackle the issue

Algorithm 1 The pseudocode of our local time-decoupled squared W_2 method for training the SNN (the local time-decoupled squared W_2 loss Eq. (2.4) can be replaced with other loss functions).

Given N observed time-series data $\{\boldsymbol{X}_i(t_j), t_j = j\Delta t, j = 1, \ldots, N_T\}_{i=1}^N$, the underlying dynamical system with unknown model parameters θ (such as the jump-diffusion process Eq. (2.9)), and the maximal epochs $i_{\max}$.
Initialize the SNN model in Fig. 1.
Input a scalar 1 into the SNN and evaluate the SNN N times independently (each time, the weights in the SNN are resampled), which outputs N sets of approximate parameters denoted by $\{\hat{\theta}\}_{i=1}^N$.
Generate N trajectories $\{\hat{\boldsymbol{X}}\}_{i=1}^N$ from the approximate dynamical system (such as Eq. (2.10)) with the approximate parameters $\{\hat{\theta}\}_{i=1}^N$.
while $i < i_{\max}$ **do**
  Perform gradient descent to minimize the loss function $\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}, \hat{\boldsymbol{X}})$ (Eq. (2.4)) and update the parameters (biases & means and variances of weights) in the SNN model.
  Input a scalar 1 into the updated SNN and evaluate the updated SNN N times independently, which outputs N sets of approximate parameters denoted by $\{\hat{\theta}\}_{i=1}^N$.
  Generate N trajectories from the approximate dynamical system with the approximate parameters $\{\hat{\theta}\}_{i=1}^N$.
end while
return the trained SNN model

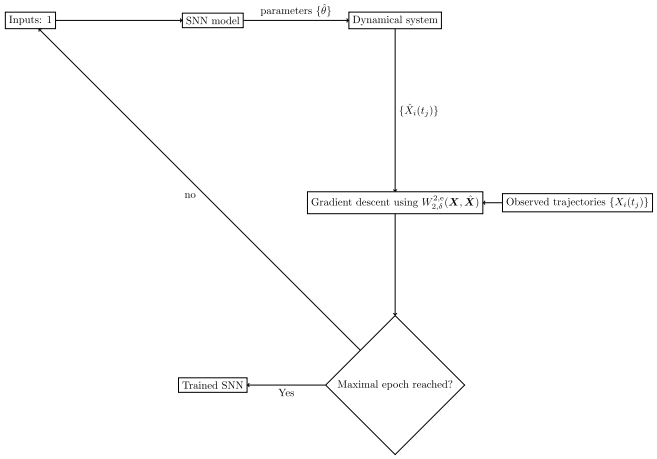

Fig. 2. A schematic diagram on the training of the SNN using our proposed local time-decoupled squared W_2 loss function.

of vanishing gradients and improve the representational power of the SNN. In the following, errors in the distribution of reconstructed model parameters denote the scaled squared W_2 distance:

$$\text{error} := \frac{W_2^2(\mu_\theta, \mu_{\hat{\theta}})}{\|\theta\|^2}, \tag{4.1}$$

where θ and $\hat{\theta}$ are unknown ground truth model parameters and reconstructed model parameters, *e.g.* in Eqs. (1.1) and (1.2), and μ_θ and $\mu_{\hat{\theta}}$ are the distributions of θ and $\hat{\theta}$, respectively. For numerically solving ODEs in all examples, we use the `odeint` function with default settings in the `torchdiffeq` package (Chen et al., 2018). In Example 4.4, we use the package developed in Xia et al. (2024a) for numerically solving a jump-diffusion process using the Euler scheme (Platen & Bruti-Liberati, 2010), which allows for back-propagation and gradient descent for hyperparameter optimization in the neural networks. The Itô integral is adopted for evaluating the stochastic integrals. The numerical experiment in Example 4.1 is conducted using Python 3.11 on a desktop with a 32-core Intel® i9-13900KF CPU (when comparing runtimes and RAM usage, we train each model on just one core). Numerical experiments in Examples 4.2–4.4 are carried out using Python 3.11 on NYU HPC with a GPU (NYU HPC, 2025).

Example 4.1. *First, we consider reconstructing the following 2D ODE characterizing the Lokta–Volterra predator-prey dynamics with one uncertain predation rate parameter:*

$$\frac{\mathrm{d}x}{\mathrm{d}t} = 2x - cxy, \quad \frac{\mathrm{d}y}{\mathrm{d}t} = \tfrac{1}{4}cxy - 2y, \\ (x(0), y(0)) = (\xi_1, \xi_2), \ c \sim \mathcal{U}(2,4), \ \xi_1, \xi_2 \sim \mathcal{U}(1,2), \ t \in [0,8]. \tag{4.2}$$

In Eq. (4.2), c is the uncertain predation rate parameter, and ξ_1, ξ_2 are two independent random variables. We train the SNN model in Fig. 1 by minimizing the loss function Eq. (J.1) (the neighborhood size $\delta = 0.4$) to reconstruct the distribution of the parameter c. For comparison, we minimize other loss functions (definitions given in Appendix J) commonly used in statistical inference tasks to train the SNN model in Fig. 1. We also compare our method with a Bayesian neural network (BNN) method used in Bayesian Neural Network Derivation (2020), Mullachery et al. (2018) which minimizes the Kullback-Leibler divergence as well as a WGAN method presented in Arjovsky et al. (2017), Gulrajani et al. (2017). For implementing the WGAN method, the generator is the same as the SNN in Fig. 1 when utilizing other loss functions, while the discriminator is a feedforward neural network with one hidden layer equipped with 32 neurons and the ReLU activation function. Within each epoch for training the generator, 256 randomly selected samples are provided to train the discriminator, repeated 4 times. After training, the generator (SNN) is used to generate the distribution of the reconstructed model parameter.

From Fig. 3(a), (b), the distribution of ground truth trajectories of the prey and predator population can be matched well by the distribution of trajectories generated by numerically solving Eq. (4.2) with $\hat{c}$ sampled from the reconstructed distribution. Furthermore, the distribution of the reconstructed $\hat{c}$ generated by the SNN model trained using our loss function Eq. (J.2) aligns well with the distribution of the ground truth predation rate c (shown in Fig. 3(c)). Using the previous time-decoupled squared W_2 loss function in Xia et al. (2024b), the MMD loss function, or the Mean^2 + Var loss function all lead to an inaccurate calculated value of the mean in the reconstructed $\hat{c}$. The WGAN method and the BNN method both yield qualitatively incorrect results. We find that training the discriminator and generator in the WGAN approach requires fine-tuning of hyperparameters by setting the learning rate to be a relatively small value of 10^{-5}, otherwise the training of the generator and discriminator will terminate prematurely due to blowup, as reported in the literature (Hoffer et al., 2017). Nonetheless, the WGAN could not yield a qualitatively correct mean of the reconstructed $\hat{c}$. On the other hand, the BNN method also does not perform well, and one possible reason could be that randomly initialized means and variances of weights as well as the biases in the neural network do not provide a good prior distribution for the BNN. Using the MSE as the loss function to train the SNN yields an almost degenerate distribution of $\hat{c}$ and the calculated value of the variance in $\hat{c}$ is inaccurate, indicating that it is not suitable to train the SNN model for reconstructing the distribution of unknown parameters. Using our proposed local time-decoupled squared W_2 loss function yields more accurate values of the mean and variance of the reconstructed $\hat{c}$ compared to other methods (shown in Fig. 3(d)). Finally, from Table 1, the runtime and RAM usage when using our proposed local time-decoupled squared W_2 loss function is generally comparable to those when using other benchmark loss functions, and the runtime and RAM usage of the WGAN method and BNN method are significantly larger than training the SNN in Fig. 1 by directly minimizing any one of the loss functions we use. Thus, using our proposed local time-decoupled squared W_2 method is more efficient than using other benchmark statistical loss functions for training the SNN in Fig. 1 to reconstruct the distribution of the unknown predation rate in Eq. (4.2), and it also outperforms the WGAN method and the BNN method.

Next, we apply our local time-decoupled squared W_2 method for the reconstruction of model parameters in a spatiotemporal PDE.

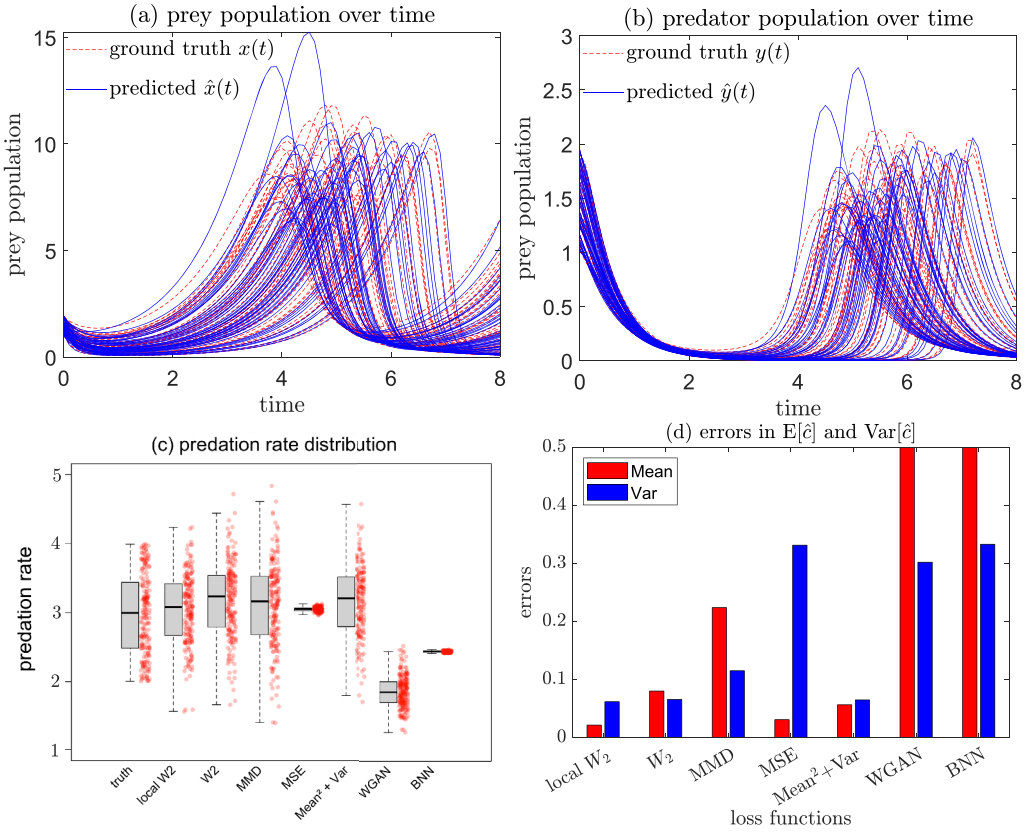

Fig. 3. (a) Ground truth (red dashed lines) prey population dynamics versus predicted prey population dynamics (blue solid lines) obtained with reconstructed predation rate $\hat{c}$. (b) Ground truth (red) predator population dynamics versus predicted predator population dynamics (blue) obtained with reconstructed predation rate $\hat{c}$. In (a) and (b), for clarity, we plot the first 50 groups of prey and predatory trajectories. Since the predation rate c in Eq. (4.2) is sampled independently for each realization of the model Eq. (4.2), the ground truth trajectories also form a distribution. (c) Ground truth $c \sim \mathcal{U}(2,4)$ versus the distribution of the approximate $\hat{c}$ when minimizing different loss functions or using the BNN or WGAN method. The black horizontal line and the box indicate the median and the interquartile range of the ground truth or predicted predation rate. (d) Errors in the predicted mean $|\mathbb{E}[\hat{c}] - \mathbb{E}[c]|$ and predicted variance $|\mathrm{Var}[\hat{c}] - \mathrm{Var}[c]|$ when minimizing different loss functions. The errors are their averaged values over 5 independent experiments. In (c) and (d), "local W_2" refers to our scaled local time-decoupled squared W_2 loss function Eq. (J.1) while "W_2" refers to previous time-decoupled squared W_2 loss function in Xia et al. (2024b).

Table 1
Computational time and memory usage when utilizing different loss functions. Mean and the standard deviation in the runtime (unit: hour) and RAM (unit: Mb) usage over five repeated experiments are recorded.

	Ours (Eq. (J.1))	Time-decoupled W_2^2	MMD	MSE	$\mathrm{Mean}^2 + \mathrm{Var}$	WGAN	BNN
Time	**1.53 ± 0.35**	1.57 ± 0.36	1.91 ± 0.47	1.37 ± 0.46	1.73 ± 0.57	7.88 ± 1.17	21.31 ± 5.40
RAM	**729.8 ± 56.3**	648.4 ± 108.7	834.3 ± 19.3	616.5 ± 110.3	614.3 ± 108.2	1341.4 ± 293.5	3406.2 ± 300.0

Example 4.2. *We consider reconstructing the distributions of parameters in the following parabolic PDE:*

$$\begin{aligned}
&\partial_t u(x,t;c_1,c_2) = \frac{c_1}{c_2^2}\partial_{xx}u(x,t;c_1,c_2) + \frac{c_1}{c_1 t+1}u(x,t;c_1,c_2),\ \ (x,t)\in\mathbb{R}\times[0,2],\\
&u(x,0) = (1+\xi)\tfrac{x}{\sqrt{4}}\cdot\exp\big(-\tfrac{x^2}{2}\big), c_1 \sim \mathcal{N}(0.5,\sigma_1^2),\\
&c_2 = \tilde{\xi} + \beta(0.5-c_1),\ \ \xi\sim\mathcal{N}(0,\sigma_3^2),\ \ \tilde{\xi}\sim\mathcal{N}(1.5,\sigma_2^2).
\end{aligned} \tag{4.3}$$

We use another parabolic PDE model to approximate Eq. (4.3):

$$\begin{aligned}
&\partial_t \hat{u}(x,t;\hat{c}_1,\hat{c}_2) = \frac{\hat{c}_1}{\hat{c}_2^2}\partial_{xx}\hat{u}(x,t;\hat{c}_1,\hat{c}_2) + \frac{\hat{c}_1}{\hat{c}_1 t+1}\hat{u}(x,t;\hat{c}_1,\hat{c}_2),\ \ (x,t)\in\mathbb{R}\times[0,2],\\
&\hat{u}(x,0) = u(x,0).
\end{aligned} \tag{4.4}$$

We wish to reconstruct the distribution of (c_1,c_2) in Eq. (4.3) using the distribution of $(\hat{c}_1,\hat{c}_2)$ in the approximate Eq. (4.4). We use a pseudo-spectral method with a spectral expansion in space to solve Eqs. (4.3) and (4.4) numerically:

$$\begin{aligned}
&u(x,t;\theta) \approx u_{n-1}(x,t;\theta) = \sum_{i=0}^{n-1} u_i(t;\theta)\hat{\mathcal{H}}_i(x),\ \ \theta := (c_1,c_2)\\
&\hat{u}(x,t;\hat{\theta}) \approx \hat{u}_{n-1}(x,t;\hat{\theta}) = \sum_{i=0}^{n-1} \hat{u}_i(t;\hat{\theta})\hat{\mathcal{H}}_i(x),\ \ \hat{\theta} := (\hat{c}_1,\hat{c}_2),
\end{aligned} \tag{4.5}$$

where $\hat{\mathcal{H}}_i$ is the generalized Hermite function described in Shen and Wang (2010).

We carry out the following sensitivity tests:

1. *Vary (σ_1,σ_2) which determines the variance in (c_1,c_2) to investigate how variances in model parameters would affect the reconstruction accuracy of the joint distribution of (c_1,c_2). Other parameters are set as $\beta = 1, \sigma_3 = 0.2, n = 12$ and $\delta = 0.1$.*
2. *Change the value of σ_3 characterizing the uncertainty in the initial condition and the size of the neighborhood δ in our loss function Eq. (J.1) to investigate how uncertainty in the initial condition and the hyperparame-*

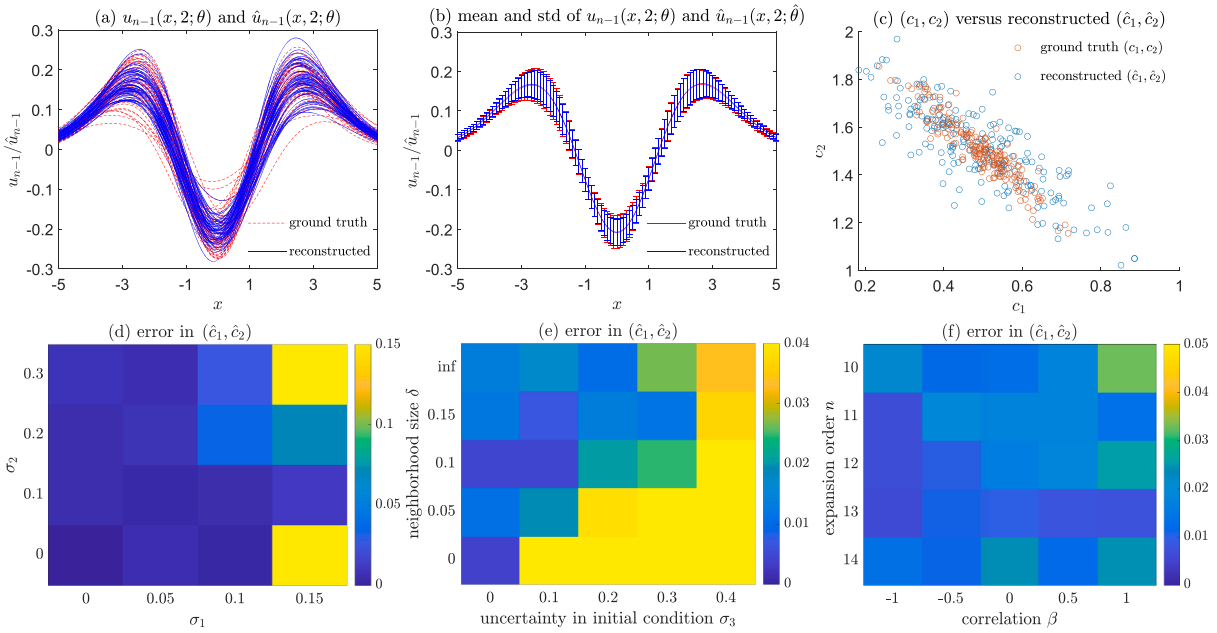

Fig. 4. (a) Ground truth $u_{n-1}(x,2;\theta)$ (red dashed lines) versus reconstructed $\hat{u}_{n-1}(x,2;\hat{\theta})$ (blue solid lines). For clarity, we only plot 50 ground truth $u_{n-1}(x,2;\theta)$ numerical solutions versus 50 approximate numerical solutions $\hat{u}_{n-1}(x,2;\hat{\theta})$ in Eq. (4.5). (b) Mean and standard deviations of the ground truth $u_{N-1}(x,2)$ versus reconstructed $\hat{u}_{n-1}(x,2)$. (c) The ground truth (c_1,c_2) versus reconstructed $(\hat{c}_1,\hat{c}_2)$ when $\beta=1,\sigma_1=0.15,\sigma_2=0.1,n=12$. In (a), (b) and (c), the parameters are $n=12$ and $\beta=1,\sigma_1=0.15,\sigma_2=0.1,\sigma_3=0.2,N=12$ and $\delta=0.1$ in the loss function Eq. (J.1). (d) Errors in $(\hat{c}_1,\hat{c}_2)$ w.r.t. different variances σ_1,σ_2 for (c_1,c_2) in Eq. (4.3) (Case 1 on Page 23). (e) Errors in $(\hat{c}_1,\hat{c}_2)$ w.r.t. different values of the variance σ_3 in the initial condition ϵ and different δ. $\delta=\inf$ indicates that we set $\delta=\infty$, which corresponds to the time-decoupled squared W_2 loss function in Xia et al. (2024a) (Case 2 on Page 23). (f) Errors in $(\hat{c}_1,\hat{c}_2)$ w.r.t. different values N and c_0 (Case 3 on Page 23).

ter δ affect the reconstruction of the distribution (c_1,c_2). Other parameters are $\beta=1,\sigma_1=0.1,\sigma_2=0.2,n=12$.

3. *Vary β which determines the correlation between c_1 and c_2 as well as the expansion order N in the spectral approximation in Eq. (4.5) to explore how the correlation between c_1 and c_2 and the dimensionality of the discretized ODE affect the reconstruction accuracy of (c_1,c_2). Other parameters are $\sigma_1=0.1,\sigma_2=0.2,\sigma_3=0.2,\delta=0.1$.*

We plot the reconstructed numerical solution $\hat{u}_{n-1}(x,t;\hat{\theta})$ versus the ground truth numerical solution $u_{n-1}(x,t;\theta)$ in Eq. (4.5) as the numerical solution to Eq. (4.3) in Fig. 4(a), (b). The distribution of ground truth numerical solutions can be matched well by the distribution of the numerical solutions generated with the reconstructed $(\hat{c}_1,\hat{c}_2)$, shown in Fig. 4(c). From Fig. 4(d), the larger the variance in c_1, the larger the errors in the predicted distribution of $(\hat{c}_1,\hat{c}_2)$. When the variance c_1 is too large, we might have close-to-zero or even negative c_1 in Eq. (4.3), which makes numerically solving Eq. (4.3) ill-posed and yields a poor reconstruction of the joint distribution of (c_1,c_2). Thus, it is reasonable to reconstruct the distribution of model parameters for well-posed dynamical systems instead of ill-posed dynamical systems. From Fig. 4(e), when the variance σ_3^2 in the initial condition is small and the initial conditions are more densely distributed, a smaller δ leads to a more accurate reconstruction of (c_1,c_2). Thus, it is necessary to consider uncertainties in the initial condition of an ODE or PDE for the accurate reconstruction of unknown model parameters and find the size of the neighborhood δ that is compatible with uncertainty in the initial state. Finally, from Fig. 4(f), the error in the reconstructed distribution of $(\hat{c}_1,\hat{c}_2)$ is independent of the dimensionality n of the discretized ODE system. Furthermore, the accuracy of the reconstructed $(\hat{c}_1,\hat{c}_2)$ is insensitive to the correlation between the two uncertain model parameters (c_1,c_2), indicating the robustness of our proposed local temporally decoupled squared W_2 method for reconstructing the distribution of (c_1,c_2) in Eq. (4.3).

As an application of our proposed method for parameter inference problems in biophysics, we reconstruct reaction rates in an 8D ODE system characterizing drug dynamics in an ocular model studied in Crawshaw et al. (2025).

Example 4.3. *We consider an ODE system that arises from modeling studies of monoclonal antibodies (MABs) injected into the vitreous gel of the eye in the treatment of wet age-related macular degeneration. The monoclonal antibodies bind to the vascular endothelial growth factor (VEGF) with this binding inhibiting the latter's stimulation of pathological capillary growth through the retina (Chappelow & Kaiser, 2008; Mitchell et al., 2018). However, injected MABs are cleared from the eye, by passing into the aqueous compartment at the front of the eye. This in turn necessitates multiple injections of MABs into the eye. Hence, analyzing how long MABs are retained within the eye and the prospect of reducing injection frequency has motivated numerous modeling studies (e.g. (Caruso et al., 2019; Crawshaw et al., 2025; Hutton-Smith et al., 2018, 2016; Lamirande et al., 2024)). However, a common theme within these studies is the need to estimate the parameters (Crawshaw et al., 2025; Hutton-Smith et al., 2016; Mitchell et al., 2018).*

Hence, we consider a simple well-mixed model in the literature for the MAB ranibizumab (Crawshaw et al., 2025; Hutton-Smith et al., 2016), with its concentration in the vitreous of the eye denoted by r_{vit}, and the vitreous VEGF concentration denoted by v_{vit}. A complex of one MAB bound to one VEGF in the vitreous has concentration c_{vit} and, noting that VEGF has two binding sites, a complex of two MABs and a VEGF is also considered, with concentration h_{vit}. The formation and disassociation of these species is represented by the law of mass action, with the reaction rates labeled by "k", and all species can pass into the aqueous at the front of the eye, with concentrations in this compartment then distinguished by the subscript "aq" ($r_{aq},v_{aq},c_{aq},h_{aq}$). Once in this latter compartment, all species are taken to pass into the bloodstream via Schlemm's canal, with a clearance rate CL. In addition, there is a flux, V_{in}, of vitreal VEGF to reflect the rate of elevated levels of VEGF leaking into the vitreous in pathology. This generates the set of eight ODEs:

Vitreous

$$\begin{aligned}
\frac{\mathrm{d}v_{vit}}{\mathrm{d}t} &= (k_{off}c_{vit} - 2k_{on}v_{vit}r_{vit}) - k_v^{el}v_{vit} + \frac{V_{in}}{V_{vit}},\\
\frac{\mathrm{d}r_{vit}}{\mathrm{d}t} &= (k_{off}c_{vit} - 2k_{on}v_{vit}r_{vit}) + (2k_{off}h_{vit} - k_{on}r_{vit}c_{vit}) - k_r^{el}r_{vit},\\
\frac{\mathrm{d}c_{vit}}{\mathrm{d}t} &= -(k_{off}c_{vit} - 2k_{on}v_{vit}r_{vit}) + (2k_{off}h_{vit} - k_{on}r_{vit}c_{vit}) - k_c^{el}c_{vit},\\
\frac{\mathrm{d}h_{vit}}{\mathrm{d}t} &= -(2k_{off}h_{vit} - k_{on}r_{vit}c_{vit}) - k_h^{el}h_{vit}.
\end{aligned} \tag{4.6}$$

Aqueous

$$\begin{aligned}
\frac{\mathrm{d}v_{aq}}{\mathrm{d}t} &= (k_{off}c_{aq} - 2k_{on}v_{aq}r_{aq}) + \frac{V_{vit}}{V_{aq}}k_v^{el}v_{vit} - \frac{CL}{V_{aq}}v_{aq},\\
\frac{\mathrm{d}r_{aq}}{\mathrm{d}t} &= (k_{off}c_{aq} - 2k_{on}v_{aq}r_{aq}) + (2k_{off}h_{aq} - k_{on}r_{aq}c_{aq}) + \frac{V_{vit}}{V_{aq}}k_r^{el}r_{vit}\\
&\quad - \frac{CL}{V_{aq}}r_{aq},\\
\frac{\mathrm{d}c_{aq}}{\mathrm{d}t} &= -(k_{off}c_{aq} - 2k_{on}v_{aq}r_{aq}) + (2k_{off}h_{aq} - k_{on}r_{aq}c_{aq}) + \frac{V_{vit}}{V_{aq}}k_c^{el}c_{vit}\\
&\quad - \frac{CL}{V_{aq}}c_{aq},\\
\frac{\mathrm{d}h_{aq}}{\mathrm{d}t} &= -(2k_{off}h_{aq} - k_{on}r_{aq}c_{aq}) + \frac{V_{vit}}{V_{aq}}k_h^{el}h_{vit} - \frac{CL}{V_{aq}}h_{aq},\ t \in [0,2]\ (\textit{unit: day}).
\end{aligned} \tag{4.7}$$

From (Crawshaw et al., 2025), we set the vitreous and aqueous humor volumes $V_{vit} = 2.05mL$ and $V_{aq} = 0.105mL$, respectively, and $V_{in} = 5.408pmol \cdot day^{-1}$. We aim to reconstruct the distribution of the seven reaction rate parameters: $k_{off}, k_{on}, k_v^{el}, k_r^{el}, k_c^{el}, k_h^{el}$ and CL in Eqs. (4.6) and (4.7), which are subject to uncertainties in the drug properties (Crawshaw et al., 2025; Hutton-Smith et al., 2016; Mitchell et al., 2018).

We generate a synthetic data set of model parameters by sampling the seven kinetic parameters: $\boldsymbol{k} := (k_{off}, k_{on}, k_v^{el}, k_r^{el}, k_c^{el}, k_h^{el}, CL)$ from the following model:

$$\boldsymbol{k} = \boldsymbol{k}_0 + c\boldsymbol{k}_0 * A\tilde{\boldsymbol{k}}, \tag{4.8}$$

where

$$\begin{aligned}
\boldsymbol{k}_0 = (&1.669day^{-1}, 0.00114pM^{-1}\cdot day^{-1}, 0.575day^{-1}, 0.293day^{-1},\\
&0.259day^{-1}, 0.176day^{-1}, 2.505mL\cdot day^{-1})
\end{aligned} \tag{4.9}$$

is the vector of mean values of those kinetic parameters used in Crawshaw et al. (2025). In Eq. (4.8), $*$ is the Hadamard componentwise product. $A \in \mathbb{R}^{7\times7}$ is a randomly generated matrix whose components are sampled independently from the distribution $\mathcal{U}(-\frac{1}{2}, \frac{1}{2})$. In Eq. (4.8), $\tilde{\boldsymbol{k}} := (k_1, k_2, \ldots, k_7)$ is independently generated for each trajectory in the training dataset. Omitting the units for simplicity, we sample $k_1, k_2 \sim \mathcal{U}(0,1)$, $k_3, k_4 \sim \mathcal{N}(0, 0.5^2)$, $k_5 \sim Exp(2)$, $k_6 \sim B(2,5)$ (the Beta distribution with shape parameters $\alpha = 2, \beta = 5$), and $k_7 \sim \Gamma(2,2)$ (the Gamma distribution with a shape parameter $\alpha = 2$ and scale parameter $\lambda = 2$). The initial condition of each trajectory is independently sampled from $\mathcal{N}(\boldsymbol{I}_7, 0.05^2 I_{7\times7})$, where $\boldsymbol{I}_7 \in \mathbb{R}^7$ refers to a constant vector whose components are 1.

The distribution of trajectories of the four quantities $v_{vit}(t), r_{vit}(t), c_{vit}(t)$ and $h_{vit}(t)$ obtained with the parameter vector $\boldsymbol{k}$ sampled from the ground truth distribution Eq. (4.8) can be matched well by the distribution of trajectories obtained by using the parameter vector $\boldsymbol{k}$ sampled from the reconstructed distribution generated by the trained SNN in Fig. 5(a)–(d). Additionally, we plot the empirical joint distribution of any two parameters in $\boldsymbol{k}$ versus the empirical joint reconstructed distribution of the corresponding two variables in Fig. 5(e), and most pairwise joint distributions of any two components in $\boldsymbol{k}$ can be matched well by their reconstructed counterparts. However, the reconstruction of k_{on} is not accurate. A possible reason could be that the magnitude of k_{on} ($O(10^{-3})$) is much smaller than that of other parameters ($O(1)$), making it harder to reconstruct the distribution of k_{on}. Another possibility is a lack of practical identifiability for these parameters. Since only observed trajectories are available, the reconstruction of the distribution of model parameters using our method signifies a "worst case scenario" with no prior information on the magnitudes of model parameters.

Table 2
Errors in the reconstructed distribution of kinetic parameters in the ocular pharmacokinetic model. Here, ELU refers to using the ELU activation function for forward propagation and ResNet refers to using the ResNet technique (described in Fig. 1). When using the W_2, MMD, MSE, and the Mean^2 + Var loss functions, the SNN has 3 hidden layers with 10 neurons in each layer.

Width	# of layers	Forward propagation	Initialization for weights & biases	Error
10	1	ELU	$\mathcal{N}(0, 0.03^2)$	7.79×10^{-3}
10	2	ELU	$\mathcal{N}(0, 0.03^2)$	1.39×10^{-3}
10	3	ELU	$\mathcal{N}(0, 0.03^2)$	1.10×10^{-3}
10	4	ELU	$\mathcal{N}(0, 0.03^2)$	8.93×10^{-4}
5	3	ELU	$\mathcal{N}(0, 0.03^2)$	8.85×10^{-4}
15	3	ELU	$\mathcal{N}(0, 0.03^2)$	9.20×10^{-4}
20	3	ELU	$\mathcal{N}(0, 0.03^2)$	0.156
10	2	ResNet	$\mathcal{N}(0, 0.03^2)$	8.25×10^{-4}
10	3	ResNet	$\mathcal{N}(0, 0.03^2)$	8.67×10^{-4}
10	4	ResNet	$\mathcal{N}(0, 0.03^2)$	8.71×10^{-4}
10	3	ELU	$\mathcal{N}(0, 0)$	0.609
10	3	ELU	$\mathcal{N}(0, 0.01^2)$	7.98×10^{-4}
10	3	ELU	$\mathcal{N}(0, 0.02^2)$	8.42×10^{-4}
W_2	3	ELU	$\mathcal{N}(0, 0.03^2)$	1.24×10^{-3}
MMD	3	ELU	$\mathcal{N}(0, 0.03^2)$	1.21×10^{-3}
MSE	3	ELU	$\mathcal{N}(0, 0.03^2)$	1.23×10^{-3}
Mean^2 + Var	3	ELU	$\mathcal{N}(0, 0.03^2)$	1.19×10^{-3}

We also investigate how the number of neurons in each layer, the number of hidden layers in the SNN model (Fig. 1), the initialization for the distribution of weights in the SNN model, as well as whether adopting the ResNet technique (He et al., 2016) for forward propagation would affect the accuracy of the reconstructed distribution of the kinetic parameters. From Table 2, SNNs with more than one hidden layer and 10 neurons in each layer can all reconstruct the distribution of $\boldsymbol{k}$ in Eq. (4.8) well. Thus, we do not need a wide or deep SNN model in Fig. 1 to reconstruct the distribution of $\boldsymbol{k}$. As an additional comparison with other loss functions, we train the SNN with 3 hidden layers and 10 neurons in each layer by minimizing the W_2, MMD, MSE, and the Mean^2 + Var loss functions (defined in Appendix J) as was done in Example 4.1. Our results indicate that the SNN model, when equipped with appropriate numbers of hidden layers and neurons in each layer, has the ability to approximate the distribution of unknown model parameters well, and minimizing our local time-decoupled squared W_2 loss function can most efficiently train the SNN and yield the most accurate reconstructed distribution of unknown model parameters. Also, applying the ResNet technique can moderately increase the reconstruction accuracy when the number of hidden layers increases. Finally, initializing the weights and biases to small non-zero values is important, as the reconstruction error is huge if we set all weights and biases to zero. It is worth further investigation on how to optimally design the structure of the SNN model in Fig. 1, which is beyond the scope of this paper.

Finally, we consider reconstructing the distribution of parameters of a jump-diffusion process, in which intrinsic stochasticity from the Wiener process and Poisson process as well as uncertainty in the initial condition of the jump-diffusion process are coupled with uncertainty in the parameters governing the jump-diffusion process.

Example 4.4. *We reconstruct the distribution of uncertain parameters governing a jump-diffusion process that can be used to describe the posited stock returns (Merton, 1976; Xia et al., 2024a). Instead of considering a deterministic jump magnitude function as in (Xia et al., 2024a, Example 2), we consider the following jump-diffusion process:*

$$\begin{aligned}
&\mathrm{d}X_t = 0.05\mathrm{d}t + s\sqrt{|X_t|}\mathrm{d}B_t + \int_U \xi X_t \mathrm{d}\tilde{N}(\gamma(\mathrm{d}\xi)\mathrm{d}t),\quad t \in [0,2],\\
&s \sim \sigma_0\mathcal{N}(1,1),\ \xi \sim \mathcal{N}(\beta_0, \sigma_1^2),\ X_0 \sim \mathcal{N}(2, \sigma_2^2).
\end{aligned} \tag{4.10}$$

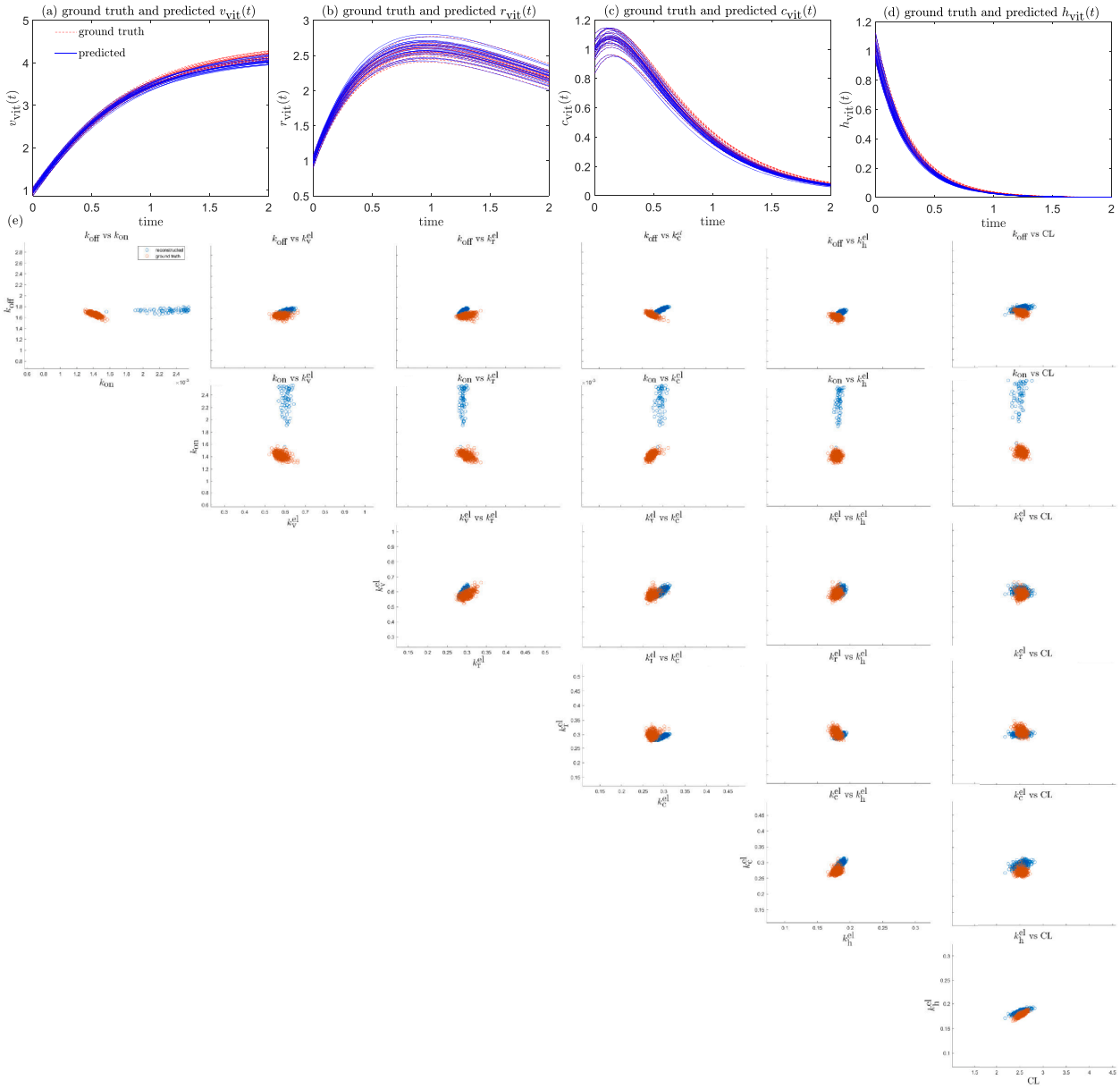

Fig. 5. (a)–(d) The first 50 out of 400 trajectories of the four quantities $v_{\text{vit}}(t), r_{\text{vit}}(t), c_{\text{vit}}(t), h_{\text{vit}}(t)$ obtained with the parameter vector $\boldsymbol{k}$ sampled from the ground truth distribution Eq. (4.8) (red dashed lines) versus trajectories obtained by using the parameter vector $\boldsymbol{k}$ sampled from the reconstructed distribution generated by the trained SNN (blue solid lines). The SNN has 3 hidden layers and 10 neurons in each layer. The nodes and weights are initialized by independently sampling from $\mathcal{N}(0, 0.03^2)$. (e) The reconstructed joint distribution of any two kinetic parameters in Eq. (4.8). In all subplots, the red dots are sampled from the ground truth joint distribution while the blue dots are sampled from the reconstructed distribution. When using the ResNet technique for forward propagation, the scalar input of the SNN is 0.1 as inputting 1 leads to overflow during training.

In Eq. (4.10), the drift term represents the risk-free interest rate, the diffusion term stands for the fluctuation in the stock price, and the jump term represents events of paying dividends. In Eq. (4.10), $\tilde{N}$ is a compensated Poisson process defined in Eq. (2.11). We aim at reconstructing the distributions of s as well as ξ employing two separate SNNs using the distribution of $\hat{s}_0$ and $\hat{\xi}$ in the following approximate jump-diffusion process:

$$\mathrm{d}\hat{X}_t = 0.05\mathrm{d}t + \hat{s}\sqrt{|\hat{X}_t|}\mathrm{d}B_t + \int_U \hat{\xi}\hat{X}_t \mathrm{d}\hat{N}(\gamma(\mathrm{d}\xi)\mathrm{d}t),$$
$$\hat{X}_0 = X_0,\ t \in [0,2]. \tag{4.11}$$

In Eq. (4.11), $\hat{N}$ is another compensated Poisson process that is independent of $\tilde{N}$. It has been showned in Xia et al. (2024a) that when $\xi \equiv 1$, larger values of s and β_0 make it more difficult to reconstruct the jump-diffusion process i.e., relative errors in the learned diffusion and jump functions get larger because the trajectories are more sparsely distributed. Here, we carry out further experiments on the following cases:

1. *Change the values of β_0 as well as the value of σ_1 in Eq. (4.10) to explore how the mean and variance of the jump magnitude function affect the reconstruction of distributions of $|s|$ and ξ (here we reconstruct the distribution of $|s|$ because $\hat{s}\sqrt{|\hat{X}_t|}\mathrm{d}B_t$ and $|\hat{s}|\sqrt{|\hat{X}_t|}\mathrm{d}B_t$ are identically distributed). Other parameters are $\sigma_0 = 0.3, \sigma_2 = 0.1$, and $\delta = 0.1$ (the neighborhood size in the loss function Eq. (J.1)).*
2. *Vary the value of σ_0 as well as the value of σ_1 to investigate how the uncertainty in the diffusion function and uncertainty in the jump magnitude affect the reconstruction of distributions of $|s|$ and ξ. Other parameters are $\beta_0 = 0.3, \sigma_2 = 0.1$, and $\delta = 0.1$.*

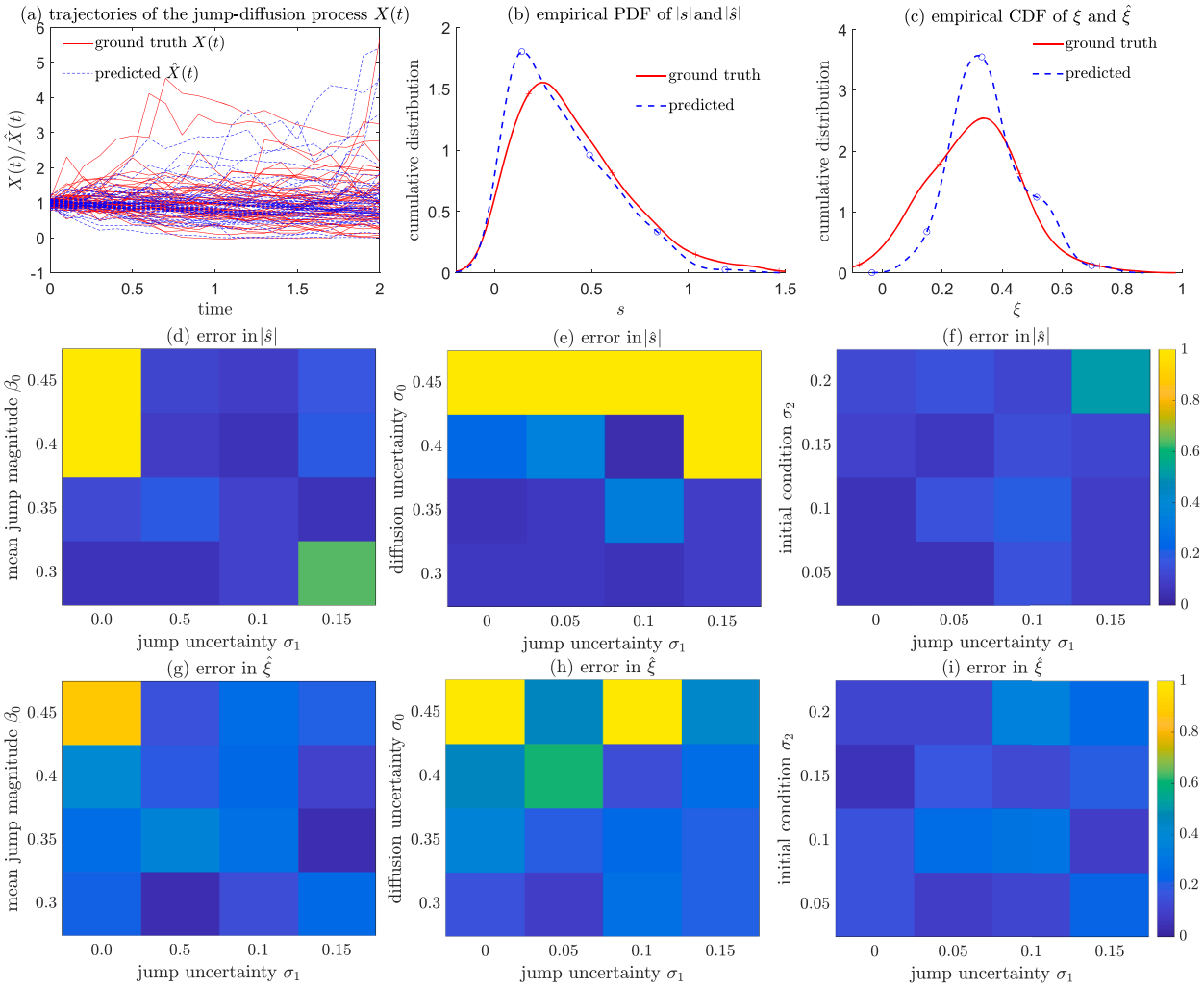

Fig. 6. (a) Ground truth trajectories generated from Eq. (4.10) versus the reconstructed trajectories generated from the approximate Eq. (4.11). For clarity, we plot 50 ground truth trajectories and 50 reconstructed trajectories. (b) The empirical probability density function ground truth $|s|$ versus the empirical probability density function of the reconstructed $|\hat{s}|$ in Eqs. (4.10) and (4.11). (c) The empirical probability density function ground truth ξ versus the empirical probability density function of the reconstructed $|\hat{\xi}|$ in Eqs. (4.10) and (4.11). In (a)–(c), $\sigma_0 = 0.3, \beta_0 = 0.35, \sigma_1 = 0.15, \sigma_2 = 0.1$, and $\delta = 0.1$ in the loss function Eq. (J.1). (d) and (g) The errors in the reconstructed distribution of $\hat{\sigma}$ and ξ for case 1 of Example 4.4 on Page 11, respectively. (e) and (h) The errors in the reconstructed distribution of $\hat{\sigma}$ and ξ for case 2 of Example 4.4 on Page 12, respectively. (f) and (i) The errors in the reconstructed distribution of $\hat{\sigma}$ and ξ for case 3 of Example 4.4 on Page 12, respectively.

3. *Vary the value of σ_1 and σ_2 to determine how uncertainty in the initial condition and jump magnitude affects the reconstruction of $|s|$ and ξ. Other parameters are $\sigma_0 = 0.3, \beta_0 = 0.3, \sigma_2 = 0.1$, and $\delta = 0.1$.*

From Fig. 6(a), the distribution of reconstructed trajectories of the approximate jump-diffusion process Eq. (4.11) match well with the distribution of ground truth jump-diffusion trajectories at each time t. Also, from Fig. 6(b), (c), the probability density functions of $|s|$ and ξ can be matched well by the probability density functions of $|\hat{s}|$ and $\hat{\xi}$ in Eq. (4.11), respectively. From Fig. 6(d), (g), when the mean and variance of ξ in the jump function of Eq. (4.10) become large, the reconstruction of the distributions of s and ξ is also less accurate. The increase in the mean β_0 of the jump magnitude impacts the reconstruction accuracy more than the increase in the variance σ_1 does. From Fig. 6(e), (h), a larger σ_0 characterizing greater uncertainty in the diffusion function of the jump-diffusion process Eq. (4.10) also leads to less accurate reconstructions of both s and ξ. Finally, from Fig. 6(f), (i), choosing a neighborhood size $\delta = 0.1$ works well for $\sigma_2 \in [0.05, 0.2]$ characterizing different levels of uncertainty in the initial condition and errors in both $|\hat{s}|$ and $\hat{\xi}$ are well controlled.

When the form of the diffusion function in the jump-diffusion process in Eq. (4.10) is unknown but the diffusion function itself is deterministic, we can use a deterministic parameterized neural network to reconstruct the diffusion function as was done in Xia et al. (2024a) while simultaneously using an SNN to reconstruct the distribution of ξ in the jump function of Eq. (4.10). When the diffusion function is deterministic, the reconstruction accuracy of both the diffusion function and the distribution of ξ in Eq. (4.10) is good. The results are shown in Appendix K.

5. Summary and conclusions

In this paper, we proposed and analyzed a local time-decoupled squared W_2 distance method for reconstructing the distributions of parameters in specific dynamical systems from time-series data using an SNN model (Fig. 1). We also analyzed the SNN model and proved that it could approximate any continuous random variable as long as moderate assumptions are satisfied, making it a suitable model for reconstructing the distribution of parameters in a dynamical system. Our method took advantage of a previous local squared W_2 method

and a previous time-decoupled squared W_2 method so that both uncertainty in the initial state and intrinsic fluctuations such as the Wiener process in the dynamics could be considered. We showcased the effectiveness of our approach for reconstructing the distribution of model parameters in several dynamical systems such as ODEs, PDEs and SDEs. Our method outperformed several other benchmark statistical methods.

One limitation of our proposed method is that when the model parameters span across different magnitudes, the reconstruction of the distribution of model parameters whose magnitudes are smaller is less accurate. In practice, prior information on the distribution of model parameters might be available either from biophysical estimates or experimental assays. Thus, how to incorporate prior information of the model parameter distribution, such as confining the range of model parameters, is a potential future research direction. Specifically, it is illuminating to make comparisons with Bayesian approaches or Monte–Carlo simulation approaches, both of which can incorporate a prior distribution of model parameters, in terms of both accuracy and computational complexity. Also, it is helpful to consider applying our method to quantify uncertainties in model selection from time-series data or spatiotemporal data (Nardini et al., 2020). Taking into account measurement errors might also be necessary when such errors are not negligible (Nardini & Bortz, 2019). The computational complexity of evaluating our proposed local time-decoupled squared W_2 loss function is $O(N_T N\mathbf{E}[(N^{\#}(\boldsymbol{X}_0;\delta))^3 \log\left(N^{\#}(\boldsymbol{X}_0;\delta)\right)])$, where $N^{\#}(\boldsymbol{X}_0;\delta)$ refers to the number of trajectories in the data set such that their initial conditions satisfy $\|\boldsymbol{X}(0)-\boldsymbol{X}_0\|\le\delta$ and N_T denotes the number of time steps. Therefore, considering utilizing entropic regularized Wasserstein distances and applying the Sinkhorn algorithm (Cuturi, 2013) to solve the corresponding optimal transport problems could be helpful in further reducing the computational complexity to $O(N_T N\mathbf{E}[(N^{\#}(\boldsymbol{X}_0;\delta))^2])$. Additionally, it will be promising to apply our proposed local time-decoupled squared W_2 method for reconstructing the distribution of uncertain parameters in more complicated dynamical systems such as multidimensional SPDEs. It will be useful to analyze other stochastic neural networks' ability to approximate the distribution of uncertain model parameters, especially model parameters that take categorical values. On the other hand, more theoretical analysis on the Wasserstein-distance-based loss function, such as proving that minimizing a Wasserstein-distance-type loss function is sufficient for reconstructing the distribution of parameters in a dynamical system, can also be interesting, which would require the analysis of identifiability of model parameters, i.e., whether two different sets of model parameters would lead to different trajectories. Finally, more theoretical and empirical studies on how to design the optimal architecture, *e.g.* the depth and width, of the SNN model in Fig. 1 would be informative.

CRediT authorship contribution statement

Mingtao Xia: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Project administration, Methodology, Investigation, Formal analysis, Conceptualization; **Qijing Shen:** Writing – review & editing, Visualization, Software, Methodology, Investigation; **Philip K. Maini:** Writing – review & editing, Resources, Investigation; **Eamonn A. Gaffney:** Writing – review & editing, Resources, Investigation; **Alex Mogilner:** Writing – review & editing, Resources, Investigation.

Data availability

All codes used in this manuscript are publicly available on GitHub at https://github.com/mtxia99/noisy_dynamical_system. In this study, no data were created or used.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported in part through the NYU IT High Performance Computing resources, services, and staff expertise. The authors also acknowledge Dr. Jessica R. Crawshaw for helpful discussions.

Appendix A. The proof of Theorem 2.1

Here, we prove Theorem 2.1. Note that

$$W_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}(t_i),\hat{\boldsymbol{X}}(t_i))=\frac{1}{N}\sum_{j=1}^{N}W_2^2\Big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i)\Big),\tag{A.1}$$

where N is the number of observed trajectories and $\boldsymbol{X}_{0,j}$ denotes the initial condition of the jth trajectory. Suppose $0=t_0^1<t_1^1<\ldots<t_{n_1}^1=T$; $0=t_0^2<t_1^2<\ldots<t_{n_2}^2=T$ are sets of grids on $[0,T]$. We define a third set of grids $0=t_0^3<\ldots<t_{n_3}^3=T$ such that $\{t_0^1,\ldots,t_{n_1}^1\}\cup\{t_0^2,\ldots,t_{n_2}^2\}=\{t_0^3,\ldots,t_{n_3}^3\}$. Let $\Delta t:=\max\{\max_i(t_{i+1}^1-t_i^1),\max_j(t_{j+1}^2-t_j^2),\max_k(t_{k+1}^3-t_k^3)\}$. We denote $\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t)$ and $\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t)$ to be the empirical conditional probability distributions of $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ at time t conditioned on $|\boldsymbol{X}(0)-\boldsymbol{X}_{0,j}|\le\delta$ and $|\hat{\boldsymbol{X}}(0)-\boldsymbol{X}_{0,j}|\le\delta$, respectively. We shall show that:

$$\Big|\sum_{i=0}^{n_1-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{i=0}^{n_3-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^3),\hat{\boldsymbol{X}}(t_i^3)\big)(t_{i+1}^3-t_i^3)\Big|\to 0,\tag{A.2}$$

as $\Delta t\to 0$.

First, suppose in the interval (t_i^1,t_{i+1}^1), we have $t_i^1=t_\ell^3<t_{\ell+1}^3<\ldots<t_{\ell+s}^3=t_{i+1}^1$, $s\ge 1$. For $s>1$, since $t_{i+1}^1-t_i^1=\sum_{k=\ell}^{\ell+s-1}(t_{k+1}^3-t_k^3)$, we have:

$$\begin{aligned}&\Big|W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{k=\ell}^{\ell+s-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_k^3),\hat{\boldsymbol{X}}(t_k^3)\big)(t_{k+1}^3-t_k^3)\Big|\\&\le\frac{1}{N}\sum_{j=1}^{N}\sum_{k=\ell+1}^{\ell+s-1}\Big(W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)+W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)\Big)\\&\quad\times\Big|W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)-W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)\Big|(t_{k+1}^3-t_k^3).\end{aligned}\tag{A.3}$$

Since $\|\boldsymbol{X}\|$ and $\|\hat{\boldsymbol{X}}\|$ are uniformly bounded, we have

$$W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)\le\sup_{\boldsymbol{X}_0}\mathbb{E}[\|\boldsymbol{X}(t)\|^2]^{\frac{1}{2}}+\mathbb{E}[\|\hat{\boldsymbol{X}}(t)\|^2]^{\frac{1}{2}}\le X+\hat{X}\tag{A.4}$$

and

$$W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)\le X+\hat{X},\tag{A.5}$$

where $X,\hat{X}$ are the upper bounds in Eq. (2.7). We can take a specific coupling measure $\pi^*_{\delta,\boldsymbol{X}_{0,j}}\big(\boldsymbol{X}(t_i^1),\boldsymbol{X}(t_k^3)\big)$ such that if we regard

$$\big(\boldsymbol{X}(t_i^1),\boldsymbol{X}(t_k^3)\big):\mathbb{R}^{2d}\to\mathbb{R}^{2d}\tag{A.6}$$

as a mapping from initial state space $(\boldsymbol{X}_0,\tilde{\boldsymbol{X}}_0)\in\mathbb{R}^{2d}$ to the solutions $(\boldsymbol{X}(t_i^1),\tilde{\boldsymbol{X}}(t_k^3))\in\mathbb{R}^{2d}$ at times (t_i^1,t_k^3) satisfying $\boldsymbol{X}(t_i^1)=\boldsymbol{X}_0,\tilde{\boldsymbol{X}}(t_k^3)=\tilde{\boldsymbol{X}}_0$, then $\pi^*_{\delta,\boldsymbol{X}_{0,j}}$ is the pushforward measure of the probability measure of the initial condition $\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}\cdot\delta_{\tilde{\boldsymbol{X}}_0=\boldsymbol{X}_0}$ under the mapping $(\boldsymbol{X}(t_i^1),\boldsymbol{X}(t_i^3))$. Here,

$\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}$ is the empirical conditional distribution of the initial condition $\boldsymbol{X}_0$ conditioned on $|\boldsymbol{X}_{0,j}-\boldsymbol{X}_0|\le\delta$. Then, for any $\boldsymbol{X}_{0,j}$, we have

$$\begin{aligned}W_2^2(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j}}(t_i^1),\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j}}(t_k^3)) &\le \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}_{(\boldsymbol{X}(t_i^1),\boldsymbol{X}(t_k^3))\sim\pi^*_{\delta,\boldsymbol{X}_{0,j}}}\left[\|\boldsymbol{X}(t_k^3)-\boldsymbol{X}(t_i^1)\|_2^2\right]\\ &\le \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i^1}^{t_{i+1}^1}\sum_{i=1}^d f_i^2(\boldsymbol{X}(t),t;\theta)\mathrm{d}t\right](t_{i+1}^1-t_i^1).\end{aligned}\tag{A.7}$$

Similarly, we have

$$W_2^2\big(\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j}}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j}}(t_k^3)\big)\le\sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t),t;\hat{\theta})\mathrm{d}t\right](t_{i+1}^1-t_i^1).\tag{A.8}$$

Using the triangular inequality of the Wasserstein distance (Clement & Desch, 2008), we have

$$\begin{aligned}&\left|W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)-W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)\right|\\ &\quad\le\left|W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)-W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^1)\big)\right|\\ &\qquad+\left|W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)-W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)\right|\\ &\quad\le W_2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)+W_2\big(\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big).\end{aligned}\tag{A.9}$$

Substituting Eqs. (A.4), (A.7), (A.8), and (A.9) into Eq. (A.3), we conclude that

$$\begin{aligned}&\left|W_2^2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)(t_{i+1}^1-t_i^1)\right.\\ &\quad\left.-\sum_{k=\ell}^{\ell+s-1}W_2^2\big((\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)(t_{k+1}^3-t_k^3)\right|\\ &\le 2(X+\hat{X})(t_{i+1}^1-t_i^1)\big(\sqrt{F_i\Delta t}+\sqrt{\hat{F}_i\Delta t}\big),\end{aligned}\tag{A.10}$$

where

$$\begin{aligned}F_i &:= \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t),t;\theta)\mathrm{d}t\right],\\ \hat{F}_i &:= \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t),t;\hat{\theta})\mathrm{d}t\right].\end{aligned}\tag{A.11}$$

Summing over i and j for the inequality (A.10), we have:

$$\begin{aligned}&\left|\sum_{i=0}^{n_1-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{i=0}^{n_3-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_k^3),\hat{\boldsymbol{X}}(t_k^3)\big)(t_{k+1}^3-t_k^3)\right|\\ &\quad\le 2(X+\hat{X})T\max_i\big(\sqrt{F_i\Delta t}+\sqrt{\hat{F}_i\Delta t}\big).\end{aligned}\tag{A.12}$$

Similarly,

$$\begin{aligned}&\left|\sum_{i=0}^{n_2-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^2),\hat{\boldsymbol{X}}(t_i^2)\big)(t_{i+1}^2-t_i^2)-\sum_{i=0}^{n_2-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_k^3),\hat{\boldsymbol{X}}(t_k^3)\big)(t_{k+1}^3-t_k^3)\right|\\ &\quad\le 2(X+\hat{X})T\max_i\left(\sqrt{F_i'\Delta t}+\sqrt{\hat{F}_i'\Delta t}\right),\end{aligned}\tag{A.13}$$

where

$$\begin{aligned}F_i' &:= \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i^2}^{t_{i+1}^2}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t),t;\hat{\theta})\mathrm{d}t\right],\\ \hat{F}_i' &:= \sup_{\boldsymbol{X}_{0,j}}\mathbb{E}\left[\int_{t_i^2}^{t_{i+1}^2}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t),t;\hat{\theta})\mathrm{d}t\right].\end{aligned}\tag{A.14}$$

Thus, as $\Delta t\to 0$,

$$\left|\sum_{i=0}^{n_1-1}W_{2,\delta}^{2,\mathrm{e}}\big(\nu(t_i^1),\hat{\nu}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{i=0}^{n_2-1}W_{2,\delta}^{2,\mathrm{e}}\big(\nu(t_i^2),\hat{\nu}(t_i^2)\big)(t_{i+1}^2-t_i^2)\right|\to 0,\tag{A.15}$$

which implies the limit

$$\lim_{\max(t_{i+1}^1-t_i^1)\to 0}\sum_{i=0}^{N-1}W_{2,\delta}^{2,\mathrm{e}}\big(\nu(t_i^1),\hat{\nu}(t_i^1)\big)(t_i^1-t_{i-1}^1)\tag{A.16}$$

exists. Therefore, the local time-decoupled squared W_2 distance in Eq. (2.6):

$$\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}}) := \int_0^T W_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}(t),\hat{\boldsymbol{X}}(t))\mathrm{d}t\tag{A.17}$$

is well-defined.

Appendix B. Proof of Corollary 2.1

The proof of Corollary 2.1 is similar to the proof of Theorem 2.1 and the proof of Theorem 3.1 in Xia et al. (2024a). Suppose $0=t_0^1<t_1^1<\ldots<t_{n_1}^1=T$; $0=t_0^2<t_1^2<\ldots<t_{n_2}^2=T$ are two sets of grids on $[0,T]$. We define a third set of grids $0=t_0^3<\ldots<t_{n_3}^3=T$ such that $\{t_0^1,\ldots,t_{n_1}^1\}\cup\{t_0^2,\ldots,t_{N_2}^2\}=\{t_0^3,\ldots,t_{n_3}^3\}$. Let $\Delta t:=\max\{\max_i(t_{i+1}^1-t_i^1),\max_j(t_{j+1}^2-t_j^2),\max_k(t_{k+1}^3-t_k^3)\}$. We denote $\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t)$ and $\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t)$ to be the empirical conditional probability distributions of $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ at time t conditioned on $|\boldsymbol{X}(0)-\boldsymbol{X}_{0,j}|\le\delta$ and $|\hat{\boldsymbol{X}}(0)-\boldsymbol{X}_{0,j}|\le\delta$, respectively. We need to show that

$$\begin{aligned}&\left|\sum_{i=0}^{n_1-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\right.\\ &\quad\left.\sum_{i=0}^{n_3-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^3),\hat{\boldsymbol{X}}(t_i^3)\big)(t_{i+1}^3-t_i^3)\right|\to 0.\end{aligned}\tag{B.1}$$

For $j=1,2$, we define:

$$\begin{aligned}F_i^j &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d f_\ell^2(\boldsymbol{X}(t^-),t^-;\theta)\mathrm{d}t\right],\\ \hat{F}_i^j &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d f_\ell^2(\hat{\boldsymbol{X}}(t^-),t^-;\hat{\theta})\mathrm{d}t\right],\\ \Sigma_i^j &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d\sum_{j=1}^m\sigma_{\ell,j}^2(\boldsymbol{X}(t^-),t^-)\mathrm{d}t\right],\\ \hat{\Sigma}_i^j &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d\sum_{j=1}^m\hat{\sigma}_{\ell,j}^2(\hat{\boldsymbol{X}}(t^-),t^-;\theta)\mathrm{d}t\right],\\ B_i^j &:= \sup_{\boldsymbol{X}_0,\theta}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d\int_U\beta_\ell^2(\boldsymbol{X}(t^-),\xi,t^-;\theta)\nu(\mathrm{d}\xi)\mathrm{d}t\right],\\ \hat{B}_i^j &:= \sup_{\boldsymbol{X}_0,\hat{\theta}}\mathbb{E}\left[\int_{t_i^j}^{t_{i+1}^j}\sum_{\ell=1}^d\int_U\hat{\beta}_\ell^2(\hat{\boldsymbol{X}}(t^-),\xi,t^-;\hat{\theta})\nu(\mathrm{d}\xi)\mathrm{d}t\right].\end{aligned}\tag{B.2}$$

Similar to the proof of Theorem 2.1, we find that:

$$\begin{aligned}&\left|W_2^2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_i^1)\big)(t_{i+1}^1-t_i^1)\right.\\ &\quad\left.-\sum_{k=\ell}^{\ell+s-1}W_2^2\big(\nu^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3),\hat{\nu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j},\delta}(t_k^3)\big)(t_{k+1}^3-t_k^3)\right|\\ &\le 2(X+\hat{X})(t_{i+1}^1-t_i^1)\left(\sqrt{F_i^1\Delta t+\Sigma_i^1+B_i^1}+\sqrt{\hat{F}_i^1\Delta t+\hat{\Sigma}_i^1+\hat{B}_i^1}\right).\end{aligned}\tag{B.3}$$

Summing over $i=0,\ldots,n_1-1$ and $j=1,\ldots,N$, we can obtain

$$\begin{aligned}&\left|\sum_{i=0}^{n_1-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{k=0}^{n_3-1}W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_k^3),\hat{\boldsymbol{X}}(t_k^3)\big)(t_{k+1}^3-t_k^3)\right|\\ &\quad\le 2(X+\hat{X})T\max_i\left(\sqrt{F_i^1\Delta t+\Sigma_i^1+B_i^1}+\sqrt{\hat{F}_i^1\Delta t+\hat{\Sigma}_i^1+\hat{B}_i^1}\right).\end{aligned}\tag{B.4}$$

Similarly, we have:

$$\Big|\sum_{i=0}^{n_2-1} W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^2),\hat{\boldsymbol{X}}(t_i^2)\big)(t_{i+1}^2-t_i^2)-\sum_{k=0}^{n_3-1} W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_k^3),\hat{\boldsymbol{X}}(t_k^3)\big)(t_{k+1}^3-t_k^3)\Big| \le 2(X+\hat{X})T\max_i\Big(\sqrt{F_i^2\Delta t+\Sigma_i^2+B_i^2}+\sqrt{\hat{F}_i^2\Delta t+\hat{\Sigma}_i^2+\hat{B}_i^2}\Big). \tag{B.5}$$

Thus,

$$\begin{aligned}\Big|\sum_{i=0}^{n_1-1} W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^1),\hat{\boldsymbol{X}}(t_i^1)\big)(t_{i+1}^1-t_i^1)-\sum_{i=0}^{n_2-1} W_{2,\delta}^{2,\mathrm{e}}\big(\boldsymbol{X}(t_i^3),\hat{\boldsymbol{X}}(t_i^2)\big)(t_{i+1}^3-t_i^3)\Big| &\le 2(X+\hat{X})T\max_i\Big(\sqrt{F_i^1\Delta t+\Sigma_i^1+B_i^1}+\sqrt{\hat{F}_i^1\Delta t+\hat{\Sigma}_i^1+\hat{B}_i^1}\Big)\\ &\quad+2(X+\hat{X})T\max_i\Big(\sqrt{F_i^2\Delta t+\Sigma_i^2+B_i^2}+\sqrt{\hat{F}_i^2\Delta t+\hat{\Sigma}_i^2+\hat{B}_i^2}\Big).\end{aligned} \tag{B.6}$$

Since $\boldsymbol{f},\boldsymbol{\sigma},\boldsymbol{\beta}$ are uniformly bounded, $F_i^j,\Sigma_i^j,B_i^j,\hat{F}_i^j,\hat{\Sigma}_i^j,\hat{B}_i^j\to 0$ as $\Delta t\to 0$ uniformly for $j=1,2$. Thus, the limit

$$\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}}) := \int_0^T W_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}(t),\hat{\boldsymbol{X}}(t))\mathrm{d}t \tag{B.7}$$

exists.

Appendix C. The proof of Theorem 2.2

In this section, we prove Theorem 2.2. First, consider $\tilde{\boldsymbol{X}}$ that solves the following jump-diffusion process:

$$\begin{aligned}&\mathrm{d}\tilde{\boldsymbol{X}}(t)=\boldsymbol{f}(\tilde{\boldsymbol{X}}(t),t;\hat{\theta})\mathrm{d}t+\boldsymbol{\sigma}(\tilde{\boldsymbol{X}}(t),t;\hat{\theta})\mathrm{d}\boldsymbol{B}_t+\int_U\boldsymbol{\beta}(\tilde{\boldsymbol{X}}(t),\xi,t;\hat{\theta})\tilde{N}(\mathrm{d}t,\nu(\mathrm{d}\xi)),\\ &\tilde{\boldsymbol{X}}(0)=\boldsymbol{X}(0).\end{aligned} \tag{C.1}$$

Using Theorem 2.1 in Xia et al. (2024a), we have:

$$\begin{aligned}&\mathbb{E}\Big[\Big\|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\Big\|^2\Big]\\ &\le\mathbb{E}\big[H(t)|\boldsymbol{X}(0)\big]\exp\Big(\big(2C+1+(2C+1)m+(2C+1)\gamma(U)\big)td\Big),\end{aligned} \tag{C.2}$$

where $\boldsymbol{X}(0)$ is the initial condition and $H(t)$ is defined as

$$\begin{aligned}H(t) :=\ &\mathbb{E}\Big[\sum_{i=1}^d\int_0^t\big(f_i(\boldsymbol{X}(s^-),s^-;\theta)-f_i(\boldsymbol{X}(s^-),s^-;\hat{\theta})\big)^2\mathrm{d}s\Big]\\ &+\mathbb{E}\Big[\sum_{i=1}^d\int_0^t\sum_{j=1}^m\big(\sigma_{i,j}(\boldsymbol{X}(s^-),s^-;\theta)-\sigma_{i,j}(\boldsymbol{X}(s^-),s^-;\hat{\theta})\big)^2\mathrm{d}s\Big]\\ &+\mathbb{E}\Big[\sum_{i=1}^d\int_0^t\int_U\big(\beta_i(\boldsymbol{X}(s^-),\xi,s^-;\theta)-\beta_i(\boldsymbol{X}(s^-),\xi,s^-;\hat{\theta})\big)^2\gamma(\mathrm{d}\xi)\mathrm{d}s\Big]\\ &\le Ct(1+m+\gamma(U))\|\theta-\hat{\theta}\|^2.\end{aligned} \tag{C.3}$$

Since the probability distribution of $\hat{\boldsymbol{X}}(t)\in\mathbb{R}^d$ is the same as the probability distribution of $\tilde{\boldsymbol{X}}(t)$ for any $t\in[0,T]$, we conclude that

$$W_2^2(\nu_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}(t))=W_2^2(\nu_{\boldsymbol{X}_0}(t),\tilde{\nu}_{\boldsymbol{X}_0}(t)), \tag{C.4}$$

where $\nu_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}(t)$, and $\tilde{\nu}_{\boldsymbol{X}_0}(t)$ are the probability distributions of $\boldsymbol{X}(t),\hat{\boldsymbol{X}}(t)$, and $\tilde{\boldsymbol{X}}(t)$ given the same initial condition $\boldsymbol{X}_0$, respectively. Given any coupled distribution of $\theta,\hat{\theta}$ denoted by $\pi(\mu,\hat{\mu})$ such that its marginal distributions coincide with the probability distributions of θ and $\hat{\theta}$, we denote $\pi^*(\boldsymbol{X}(t),\tilde{\boldsymbol{X}}(t))$ to be its pushforward probability measure for $(\boldsymbol{X}(t),\tilde{\boldsymbol{X}}(t))\in\mathbb{R}^{2d}$. We can easily check that the marginal distributions of π^* coincide with $\nu_{\boldsymbol{X}_0}(t)$ and $\tilde{\nu}_{\boldsymbol{X}_0}(t)$, respectively. From Eqs. (C.2), (C.3), we have

$$\mathbb{E}_{(\boldsymbol{X}(t),\tilde{\boldsymbol{X}}(t))\sim\pi^*}\Big[\Big\|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\Big\|^2\Big]\le Ct(1+m+\gamma(U))\mathbb{E}_{(\theta,\hat{\theta})\sim\pi(\mu,\hat{\mu})}\big[\|\theta-\hat{\theta}\|^2\big]\times\exp\Big(\big(2C+1+(2C+1)m+(2C+1)\gamma(U)\big)td\Big). \tag{C.5}$$

Taking the infimum of π over all coupled distributions of $(\theta,\hat{\theta})$ whose marginal distributions are μ and $\hat{\mu}$, respectively, we conclude that

$$W_2^2(\nu_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}(t))\ \le C_1t\exp(C_0t)W_2^2(\mu,\hat{\mu}), \tag{C.6}$$

where

$$C_0 := \Big(2C+1+(2C+1)m+(2C+1)\gamma(U)\Big)d,\ C_1 := C(1+m+\gamma(U)) \tag{C.7}$$

are two constants.

Using the triangular inequality of the W_2 distance (Clement & Desch, 2008), we have

$$\begin{aligned}W_2(\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))&\le W_2(\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t))+W_2(\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))\\ &\quad+W_2(\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))\\ &\le W_2(\nu_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}(t))+W_2(\hat{\nu}_{\boldsymbol{X}_0}(t),\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t))\\ &\quad+W_2(\nu_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t))\\ &\quad+W_2(\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))+W_2(\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t)),\end{aligned} \tag{C.8}$$

where $\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t)$ and $\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t)$ are the empirical conditional probability distributions of $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ at time t conditioned on $|\boldsymbol{X}(0)-\boldsymbol{X}_0|\le\delta$ and $|\hat{\boldsymbol{X}}(0)-\boldsymbol{X}_0|\le\delta$, respectively. For any $\theta\in\mathbb{R}^\ell$, consider

$$\begin{aligned}\mathrm{d}\boldsymbol{X}(t;\boldsymbol{X}_0)&=\boldsymbol{f}(\boldsymbol{X}(t;\boldsymbol{X}_0),t;\theta)\mathrm{d}t+\boldsymbol{\sigma}(\boldsymbol{X}(t;\boldsymbol{X}_0;\theta),t;\hat{\theta})\mathrm{d}\boldsymbol{B}_t\\ &\quad+\int_U\boldsymbol{\beta}(\boldsymbol{X}(t;\boldsymbol{X}_0),\xi,t;\theta)\tilde{N}(\mathrm{d}t,\gamma(\mathrm{d}\xi)),\\ \mathrm{d}\boldsymbol{X}(t;\boldsymbol{X}_0')&=\boldsymbol{f}(\boldsymbol{X}(t;\boldsymbol{X}_0'),t;\theta)\mathrm{d}t+\boldsymbol{\sigma}(\boldsymbol{X}(t;\boldsymbol{X}_0';\theta),t;\hat{\theta})\mathrm{d}\boldsymbol{B}_t\\ &\quad+\int_U\boldsymbol{\beta}(\boldsymbol{X}(t;\boldsymbol{X}_0'),\xi,t;\theta)\tilde{N}(\mathrm{d}t,\gamma(\mathrm{d}\xi)),\\ \boldsymbol{X}(0;\boldsymbol{X}_0)&=\boldsymbol{X}_0,\ \boldsymbol{X}(0;\boldsymbol{X}_0')=\boldsymbol{X}_0',\ \|\boldsymbol{X}_0-\boldsymbol{X}_0'\|\le\delta.\end{aligned} \tag{C.9}$$

Using the stochastic Gronwall lemma (Mehri & Scheutzow, 2019, Theorem 2.2) and (Xia et al., 2024a, Theorem 2.1), we conclude that:

$$\begin{aligned}&\mathbb{E}\Big[\Big\|\boldsymbol{X}(t;\boldsymbol{X}_0)-\boldsymbol{X}(t;\boldsymbol{X}_0')\Big\|_2^2\Big]\\ &\le\exp\Big(\big(2dC+dCm+dC\gamma(U)+1\big)t\Big)\mathbb{E}[\|\boldsymbol{X}_0-\boldsymbol{X}_0'\|^2].\end{aligned} \tag{C.10}$$

Therefore, we have:

$$W_2(\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))\le\delta\exp(\tfrac{C_0t}{2}). \tag{C.11}$$

Similarly, we conclude that:

$$W_2(\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))\le\delta\exp(\tfrac{C_0t}{2}). \tag{C.12}$$

Eq. (C.6) also holds if we replace $\hat{\nu}_{\boldsymbol{X}_0}(t)$ on the LHS of Eq. (C.6) with the empirical distribution $\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t)$ and then replace $\hat{\mu}$ with $\mu_{\boldsymbol{X}_0}^{\mathrm{e}}$ on the RHS, i.e.,

$$W_2(\nu_{\boldsymbol{X}_0}(t),\nu_{\boldsymbol{X}_0}^{\mathrm{e}}(t))\le\sqrt{C_1t}\exp(\tfrac{C_0t}{2})W_2(\mu,\mu_{\boldsymbol{X}_0}^{\mathrm{e}}). \tag{C.13}$$

Similarly,

$$W_2^2(\hat{\nu}_{\boldsymbol{X}_0}(t),\hat{\nu}_{\boldsymbol{X}_0}^{\mathrm{e}}(t))\le\sqrt{C_1t}\exp(\tfrac{C_0t}{2})W_2(\hat{\mu},\hat{\mu}_{\boldsymbol{X}_0}^{\mathrm{e}}). \tag{C.14}$$

In Eqs. (C.13) and (C.14), $\mu_{\boldsymbol{X}_0}^{\mathrm{e}}$ and $\hat{\mu}_{\boldsymbol{X}_0}^{\mathrm{e}}$ denote the empirical distributions of θ and $\hat{\theta}$, respectively. Finally, by plugging Eqs. (C.11), (C.12), (C.13), (C.14), (C.6) into Eq. (C.8), we conclude that:

$$\begin{aligned}W_2(\nu_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t),\hat{\nu}_{\boldsymbol{X}_0,\delta}^{\mathrm{e}}(t))&\le 2\delta\exp(\tfrac{C_0t}{2})\\ &\quad+\sqrt{C_1t\exp(C_0t)}\big(W_2(\mu,\mu_{\boldsymbol{X}_0}^{\mathrm{e}})+W_2(\hat{\mu},\hat{\mu}_{\boldsymbol{X}_0}^{\mathrm{e}})+W_2(\mu,\hat{\mu})\big).\end{aligned} \tag{C.15}$$

Squaring both sides of Eq. (C.15), integrating over time, and taking the expectation w.r.t. the empirical probability measure of the initial condition $\boldsymbol{X}_0$, we conclude that:

$$\mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X},\hat{\boldsymbol{X}})]\le 8T\delta^2\exp(C_0T)$$

$$+\sum_{j=1}^{N}\frac{1}{N}\frac{6C_1}{C_0}T\exp(C_0T)\big(W_2^2(\mu,\hat{\mu})$$
$$+\mathbb{E}\big[W_2^2(\mu^{\mathrm{e}}_{\boldsymbol{X}_{0,j}},\mu)\big]+\mathbb{E}\big[W_2^2(\hat{\mu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j}},\hat{\mu})\big]\big). \tag{C.16}$$

Specifically, C_0 grows linearly with the dimensionality d. Finally, taking the expectation of Eq. (C.16) and applying the empirical error bound of the squared W_2 distance in Fournier and Guillin (2015), we conclude that:

$$\sum_{j=1}^{N}\frac{1}{N}\mathbb{E}\big[W_2^2(\mu^{\mathrm{e}}_{\boldsymbol{X}_{0,j}},\mu)\big]\le C_2\mathbb{E}[h(N^{\#}(\boldsymbol{X}_0;\delta),\ell)]\Theta_6^{\frac{1}{3}},$$
$$\sum_{j=1}^{N}\frac{1}{N}\mathbb{E}\big[W_2^2(\hat{\mu}^{\mathrm{e}}_{\boldsymbol{X}_{0,j}},\hat{\mu})\big]\le C_2\mathbb{E}[h(N^{\#}(\boldsymbol{X}_0;\delta),\ell)]\hat{\Theta}_6^{\frac{1}{3}}, \tag{C.17}$$

where C_2 is a constant used in (Fournier & Guillin, 2015, Theorem 1) and h is defined in Eq. (2.19). This completes the proof of Theorem 2.2.

Appendix D. Reconstructing the distribution of parameters in an SPDE

We consider the following parabolic SPDE with a Dirichlet boundary condition studied in Grecksch and Kloeden (1996):

$$\mathrm{d}U(\boldsymbol{x},t;\theta)=A(\theta)U(\boldsymbol{x},t;\theta)+f(U(\boldsymbol{x},t;\theta);\theta)\mathrm{d}t+g(U(\boldsymbol{x},t;\theta);\theta)\mathrm{d}B_t,$$
$$\boldsymbol{x}\in D,t\in[0,T],$$
$$U(\boldsymbol{x},0;\theta)=U(\boldsymbol{x},0)\in H_0^{1,2}(\Omega),U(\boldsymbol{x},0)\sim\nu_0,\ U(\boldsymbol{x},t;\theta)=0,x\in\partial D. \tag{D.1}$$

In Eq. (D.1), B_t is a standard scalar Wiener process and D is a bounded domain in $\mathbb{R}^d$ with sufficiently smooth boundary ∂D. $H_0^{1,2}(\Omega)$ is the space of functions $U:\Omega\to\mathbb{R}$ that vanish on $\partial\Omega$ such that U and its first-order generalized derivatives belong to $L^2(\Omega)$ equipped with the norm $\|U\|_{L^2}:=\int_D U^2(\boldsymbol{x},t)\mathrm{d}\boldsymbol{x}$. ν_0 is a probability measure defined on the Sobolev space $\mathcal{B}(H_0^{1,2}(\Omega))$. A is a linear operator densely defined in $L^2(\Omega)$ such that for $U\in H^{1,2}(\Omega)$, $AU\in L^2(\Omega)$. A is strongly monotone, i.e., there is a constant $\alpha>0$ such that

$$(-AU,U)\ge\alpha\|U\|^2_{H^{1,2}},\quad\forall U\in H_0^{1,2}(\Omega), \tag{D.2}$$

where $(\cdot,\cdot)$ is the inner product. In addition, f and g, which map either $L^2(\Omega)$ or $H_0^{1,2}(\Omega)$ into itself, are formed from real-valued functions of a real variable with uniformly bounded derivatives of appropriate order.

We reconstruct the distribution of the model parameter θ in Eq. (D.1) using an approximate model

$$\mathrm{d}\hat{U}(\boldsymbol{x},t;\hat{\theta})=A(\hat{\theta})\hat{U}(\boldsymbol{x},t;\hat{\theta})+f(\hat{U}(\boldsymbol{x},t;\hat{\theta});\hat{\theta})\mathrm{d}t+g(\hat{U}(\boldsymbol{x},t;\hat{\theta});\hat{\theta})\mathrm{d}\hat{B}_t,$$
$$\boldsymbol{x}\in D,t\in[0,T], \tag{D.3}$$
$$\hat{U}(\boldsymbol{x},0)=U(\boldsymbol{x},0;\theta)=U(\boldsymbol{x},0),\ U(\boldsymbol{x},t;\hat{\theta})=0,x\in\partial D.$$

In Eq. (D.3), $\hat{B}_t$ is another standard scalar Wiener process independent of B_t in Eq. (D.1). We use the Itô-Galerkin scheme for spatial discretization of Eqs. (D.1) and (D.3):

$$\mathrm{d}U_n(t;\theta)=\big(A_n(U_n(t;\theta);\theta)+f_n(U_n(t;\theta);\theta)\big)\mathrm{d}t+g_n(U_n(t;\theta);\theta)\mathrm{d}B_t,$$
$$\mathrm{d}\hat{U}_n(t;\hat{\theta})=\big(A_n(\hat{U}_n(t;\hat{\theta});\hat{\theta})+f_n(\hat{U}_n(t;\hat{\theta});\hat{\theta})\big)\mathrm{d}t+g_n(\hat{U}_n(t;\hat{\theta});\hat{\theta})\mathrm{d}\hat{B}_t. \tag{D.4}$$

In Eq. (D.4),

$$U_n(\boldsymbol{x},t;\theta):=\sum_{j=1}^{n}u_j(t;\theta)\varphi_j(\boldsymbol{x})\in X_n,\ \hat{U}_n(\boldsymbol{x},t;\hat{\theta}):=\sum_{j=1}^{n}\hat{u}_j(t;\hat{\theta})\varphi_j(\boldsymbol{x})\in X_n \tag{D.5}$$

refer to the spectral approximations of $U(\boldsymbol{x},t;\theta)$ and $\hat{U}(\boldsymbol{x},t;\hat{\theta})$ in Eqs. (D.1) and (D.3), respectively. X_n is the n-dimensional subspace of $H_0^1(\Omega)$ spanned by the basis functions $\{\varphi_1,\dots,\varphi_n\}$. In Eq. (D.4),

$$A_n(U;\theta):=P_n\big(A(\theta)U\big),\quad f_n(U;\theta):=P_n\big(f(\theta)U\big),\quad g_n(U;\theta):=P_n\big(g(\theta)U\big),$$

where P_n denotes the projection of $L^2(\Omega)$ or $H_0^{1,2}(\Omega)$ onto X_n In the spatiotemporal SPDEs (D.1) and (D.3), we assume that for all θ, the operator $-A$ has the same set of corresponding eigenfunctions

$$-A(\theta)\varphi_j=\lambda_j(\theta)\varphi_j,\quad j=1,2,\dots,n,\ \lambda_j\le\lambda_{j+1}.$$

$\varphi_i\in H_0^{1,2}(D),i=1,2,\dots$ forms an orthonormal basis in $L^2(D)$ with $\|\varphi_j\|_{L^2}=1$ and $\lambda_j(\theta)\to\infty$ uniformly in θ as $j\to\infty$. We can prove the following result.

Corollary D.1. *We assume that $A(\theta)$, interpreted as mappings of $H_0^1(\Omega)$ into $L^2(\Omega)$ and $f(\theta)$ and $g(\theta)$, interpreted as mappings of $H_0^1(\Omega)$ into itself, are Lipschitz continuous:*

$$\|A(\theta)(U,\theta)-A(\hat{\theta})(\hat{U},\hat{\theta})\|_{L^2}\le L(\|U-\hat{U}\|_{H^{1,2}}+\|\theta-\hat{\theta}\|),$$
$$\|f(U;\theta)-f(\hat{U};\hat{\theta})\|_{H^{1,2}}\le L(\|U-\hat{U}\|_{H^{1,2}}+\|\theta-\hat{\theta}\|), \tag{D.6}$$
$$\|g(U;\theta)-g(\hat{U};\hat{\theta})\|_{H^{1,2}}\le L(\|U-\hat{U}_{H^{1,2}}\|_{H^{1,2}}+\|\theta-\hat{\theta}\|),L\le\infty.$$

Furthermore, we assume that

$$\mathbb{E}[\|\theta\|^6]\le\Theta_6,\ \mathbb{E}[\|\hat{\theta}\|^6]\le\hat{\Theta}_6. \tag{D.7}$$

Then, we have the following bound for the local time-decoupled squared distance between the probability measures of U and $\hat{U}$:

$$\begin{aligned}\tilde{W}^{2,e}_{2,\delta}(U,\hat{U})\le 3\cdot\Big(&8C_0(\beta_n;n)\delta^2T\exp(C_0(\beta_n;n)T)\\&+\frac{6C_1(\beta_n)T}{C_0(\beta_n;n)}\exp(C_0(\beta_n;n)T)\\&\times(W_2^2(\mu,\hat{\mu})+(\Theta_6^{\frac{1}{3}}+\hat{\Theta}_6^{\frac{1}{3}})2T\mathbb{E}[h(N^{\#}(\boldsymbol{U}_n(0;\theta);\delta);\ell)])\Big)\\&+3T\sup_{\theta,U(\boldsymbol{x},0;\theta)}K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta)\\&+3T\sup_{\hat{\theta},U(\boldsymbol{x},0;\hat{\theta})}K_{T,U(\cdot,0),\hat{\theta}}\lambda_{N+1}^{-1}(\hat{\theta}).\end{aligned} \tag{D.8}$$

In Eq. (D.8), $K_{T,U(\cdot,0),\theta}$ is a constant that depends on $T,U(\boldsymbol{x},0)$ and θ, while $C_i(\beta_n)$ are constants in Theorem 2.2 that grow at most linearly with n. β_n is another constant depending on the eigenvalues $\{\lambda_i\}_{i=1}^n$. The vector $\boldsymbol{U}_n(0;\theta):=(u_1(0;\theta),\dots,u_n(0;\theta))$ refers to the spectral expansion of the initial condition. $\tilde{W}^{2,e}_{2,\delta}(U,\hat{U})$ is the local time-decoupled squared distance between the probability measures of U and $\hat{U}$:

$$\tilde{W}^{2,e}_{2,\delta}(U,\hat{U}):=\int_0^T W^{2,e}_{2,\delta}\big(U(\boldsymbol{x},t;\theta),\hat{U}(\boldsymbol{x},t;\hat{\theta})\big)\mathrm{d}t, \tag{D.9}$$

and

$$W^{2,e}_{2,\delta}\big(U(\boldsymbol{x},t;\theta),\hat{U}(\boldsymbol{x},t;\hat{\theta})\big):=\int W_2^2(\nu^e_{U_0,\delta}(t),\hat{\nu}^e_{U_0,\delta}(t))\nu_0^e(\mathrm{d}U_0), \tag{D.10}$$

where $\nu_0^e(\mathrm{d}U_0)$ is the empirical distribution of the initial condition $U(\cdot,0)$, and $\nu^e_{U_0,\delta}(t)$ and $\hat{\nu}^e_{U_0,\delta}(t)$ are the empirical conditional distributions of $U(\boldsymbol{x},t;\theta)$ and $\hat{\nu}^e_{U_0,\delta}(t)$ at time t conditioned on $\|U(\boldsymbol{x},0)-U_0\|_{L^2}\le\delta$ and $\|\hat{U}(\boldsymbol{x},0)-U_0\|_{L^2}\le\delta$, respectively. In Eq. (D.10), the W_2 distance between $\nu^e_{U_0,\delta}(t)$ and $\hat{\nu}^e_{U_0,\delta}(t)$ is defined as

$$W_2(\nu^e_{U_0,\delta}(t),\hat{\nu}^e_{U_0,\delta}(t)):=\inf_{\pi(\nu,\hat{\nu})}\mathbb{E}_{(U,\hat{U})\sim\pi(\nu^e_{U_0,\delta}(t),\hat{\nu}^e_{U_0,\delta}(t))}\big[\|U-\hat{U}\|^2_{L^2}\big]^{\frac{1}{2}}. \tag{D.11}$$

Proof. First, we show that A_n is also Lipschitz continuous:

$$\begin{aligned}\|A_n(U_n;\theta)-A_n(\hat{U}_n;\hat{\theta})\|_{L^2}&=\|P_n(A(U_n;\theta)-A(\hat{U}_n;\hat{\theta}))\|_{L^2}\\&\le\|A(U_n;\theta)-A(\hat{U}_n;\hat{\theta})\|_{L^2}\\&\le L(\|U_n-\hat{U}_n\|_{H^{1,2}}+\|\theta-\hat{\theta}\|).\end{aligned} \tag{D.12}$$

Because X_n is a finite dimensional space, there exists a constant β_n depending on $\varphi_1,\dots,\varphi_n$ such that $\forall U_n\in X_n$, $\|U\|_{H^{1,2}}\le\beta_n\|U\|_{L^2}$. Thus,

$$\begin{aligned}\|A_n(U_n;\theta)-A_n(\hat{U}_n;\hat{\theta})\|_{H^{1,2}}&\le\beta_n\|A_n(U_n;\hat{\theta})-A_n(\hat{U}_n;\hat{\theta})\|_{L^2}\\&\le\beta_nL(\|U_n-\hat{U}_n\|_{L^2}+\|\theta-\hat{\theta}\|).\end{aligned} \tag{D.13}$$

Similarly, f_n and g_n can also be shown to be Lipschitz continuous in U_n and θ:

$$\begin{aligned}\|f_n(U_n;\theta)-f_n(\hat{U}_n;\hat{\theta})\|_{L^2} &\le \|f_n(U_n;\theta)-f_n(\hat{U}_n;\hat{\theta})\|_{H^{1,2}}\\ &\le \beta_n L(\|U_n-\hat{U}_n\|_{L^2}+\|\theta-\hat{\theta}\|),\\ \|g_n(U_n;\theta)-g_n(\hat{U}_n;\hat{\theta})\|_{L^2} &\le \|g_n(U_n;\theta)-g_n(\hat{\theta};\hat{U}_n)\|_{H^{1,2}}\\ &\le \beta_n L(\|U_n-\hat{U}_n\|_{L^2}+\|\theta-\hat{\theta}\|).\end{aligned} \tag{D.14}$$

From Grecksch and Kloeden (1996, Section 3), for every θ, we have

$$\mathbb{E}[\|U(\boldsymbol{x},k\Delta t;\theta)-U_n(\boldsymbol{x},k\Delta t;\theta)\|^2] \le K_{k\Delta t,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta), \tag{D.15}$$

where $K_{k\Delta t,U(\cdot,0),\theta}$ is a constant depending on time $k\Delta t$ and the initial condition $U(\cdot,0)$, and θ. Without loss of generality, we assume that $K_{k\Delta t,U(\cdot,0),\theta}$ is non-decreasing in k (otherwise we can replace $K_{k\Delta t,U(\cdot,0),\theta}$ with $\tilde{K}_{k\Delta t,U(\cdot,0),\theta} := \max_{1\le i\le k} K_{i\Delta t,U(\cdot,0),\theta}$). Given the initial condition $U(\boldsymbol{x},0)$ and $P_nU(\boldsymbol{x},0)$, we denote the probability measures of $U(\boldsymbol{x},k\Delta t;\theta)$ and $U_n(\boldsymbol{x},T;\theta)$ by $\nu_{U(\cdot,0)}(k\Delta t)$ and $\nu_{n,U_n(\cdot,0)}(k\Delta t)$, respectively. Moreover, the joint probability measure of $\big(U(\boldsymbol{x},k\Delta t;\theta),U_n(\boldsymbol{x},k\Delta t;\theta)\big)$ has marginal distributions $\nu_{U(\cdot,0)}(k\Delta t)$ and $\nu_{n,U_n(\cdot,0)}(k\Delta t)$, respectively. From Eq. (D.15), we can deduce that:

$$\begin{aligned}W_2^2(\nu_{U(\cdot,0)}(k\Delta t),\nu_{n,U_n(\cdot,0)}(k\Delta t)) &\le \mathbb{E}[\|U(\boldsymbol{x},k\Delta t;\theta)-U_n(\boldsymbol{x},k\Delta t;\theta)\|^2]\\ &\le \sup_{\theta,U(\boldsymbol{x},0)} K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta).\end{aligned} \tag{D.16}$$

Furthermore, using the definition of the local squared W_2 distance in Eq. (2.4), we have:

$$\begin{aligned}W_{2,\delta}^{2,\mathrm{e}}\big(U(\cdot,k\Delta t;\theta),U_n(\cdot,k\Delta t;\theta)\big) &\le \sup_{\theta,U(\boldsymbol{x},0)}\mathbb{E}[\|U(\boldsymbol{x},k\Delta t;\theta)-U_n(\boldsymbol{x},k\Delta t;\theta)\|^2]\\ &\le \sup_{\theta,U(\boldsymbol{x},0)} K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta).\end{aligned} \tag{D.17}$$

Similarly, we can conclude that:

$$W_{2,\delta}^{2,\mathrm{e}}\big(\hat{U}(\cdot,k\Delta t;\hat{\theta}),\hat{U}_n(\cdot,k\Delta t;\hat{\theta})\big) \le \sup_{\hat{\theta},U(\boldsymbol{x},0)} K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\hat{\theta}). \tag{D.18}$$

Given the same initial condition $U(\boldsymbol{x},0)=\hat{U}(\boldsymbol{x},0)$, using the triangle inequality of the Wasserstein distance (Clement & Desch, 2008), we have

$$\begin{aligned}W_{2,\delta}^{2,\mathrm{e}}(U(\boldsymbol{x},t;\theta),\hat{U}(\boldsymbol{x},t;\hat{\theta})) &\le 3W_{2,\delta}^{2,\mathrm{e}}\big(U(\cdot,k\Delta t;\theta),U_n(\cdot,k\Delta t;\theta)\big)\\ &\quad+3W_{2,\delta}^{2,\mathrm{e}}\big(\hat{U}(\cdot,k\Delta t;\hat{\theta}),\hat{U}_n(\cdot,k\Delta t;\hat{\theta})\big)+3W_{2,\delta}^{2,\mathrm{e}}(U_n(\boldsymbol{x},t;\theta),\hat{U}_n(\boldsymbol{x},t;\hat{\theta}))\big)\\ &\le 3W_{2,\delta}^{2,\mathrm{e}}(U_n(\boldsymbol{x},t;\theta),\hat{U}_n(\boldsymbol{x},t;\hat{\theta}))+3\sup_{\theta,U(\boldsymbol{x},0)}K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta)\\ &\quad+3\sup_{\hat{\theta},U(\boldsymbol{x},0)}K_{T,U(\cdot,0),\hat{\theta}}\lambda_{N+1}^{-1}(\hat{\theta}).\end{aligned} \tag{D.19}$$

Integrating both sides of the ineqeuality (D.19) over time, we have:

$$\begin{aligned}\tilde{W}_{2,\delta}^{2,\mathrm{e}}\big(U,\hat{U}\big) &\le 3\tilde{W}_{2,\delta}^{2,\mathrm{e}}(U_n,\hat{U}_n)+3T\sup_{\theta,U(\boldsymbol{x},0)}K_{T,U(\cdot,0),\theta}\lambda_{N+1}^{-1}(\theta)\\ &\quad+3T\sup_{\hat{\theta},U(\boldsymbol{x},0)}K_{T,U(\cdot,0),\hat{\theta}}\lambda_{N+1}^{-1}(\hat{\theta}).\end{aligned} \tag{D.20}$$

Let

$$\boldsymbol{U}_n(t;\theta) := (u_1(t;\theta),\dots,u_n(t;\theta)),\ \hat{\boldsymbol{U}}_n(t,\hat{\theta}) := (\hat{u}_1(t,\hat{\theta}),\dots,\hat{u}_n(t,\hat{\theta})) \tag{D.21}$$

be two vectors of the spectral expansion coefficients of $U_n(\boldsymbol{x},t;\theta)$ and $\hat{U}_n(\boldsymbol{x},t;\hat{\theta})$ in Eq. (D.5). We have $\|U_n(\boldsymbol{x},t;\theta)\|_{L^2}=\|\boldsymbol{U}_n(t)\|$ and $\|\hat{U}_n(\boldsymbol{x},t)\|_{L^2}=\|\hat{\boldsymbol{U}}_n(t)\|$ because $\|\varphi_i\|_{L^2}=1, i=1,\dots,n$. Furthermore, $\boldsymbol{U}_n$ and $\hat{\boldsymbol{U}}_n$ are solutions to the two SDEs:

$$\begin{aligned}\mathrm{d}\boldsymbol{U}_n &= \big(\boldsymbol{A}_n(\boldsymbol{U}_n,t;\theta)+\boldsymbol{F}_n(\boldsymbol{U}_n,t;\theta)\big)\mathrm{d}t+\boldsymbol{G}_n(\boldsymbol{U}_n,t;\theta)\mathrm{d}B_t,\\ \mathrm{d}\hat{\boldsymbol{U}}_n &= \big(\boldsymbol{A}_n(\hat{\boldsymbol{U}}_n,t;\hat{\theta})+\boldsymbol{F}_n(\hat{\boldsymbol{U}}_n,t;\hat{\theta})\big)\mathrm{d}t+\boldsymbol{G}_n(\hat{\boldsymbol{U}}_n,t;\hat{\theta})\mathrm{d}B_t\end{aligned} \tag{D.22}$$

where $\boldsymbol{A}_n(\boldsymbol{U}_n,t;\theta)$, $\boldsymbol{F}_n(\boldsymbol{U}_n,t;\theta)$ and $\boldsymbol{F}_n(\boldsymbol{G}_n,t;\theta)$ are the n-dimensional vector of the coefficients in the spectral expansions of $A_n(U_n;\theta)$, $f_n(U_n;\theta)$ and $g_n(U_n;\theta)$ in Eqs. (D.13) and (D.14), respectively. $\boldsymbol{A}_n$, $\boldsymbol{F}_N$ and $\boldsymbol{G}_n$ are also Lipschitz continuous in U_n and θ from Eqs. (D.13), (D.14). Applying Eq. (2.18) in Theorem 2.2, we obtain:

$$\begin{aligned}\mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(U_n,\hat{U}_n)] &= \mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{U}_n,\hat{\boldsymbol{U}}_n)] \le 8C_0(\beta_n,n)T\delta^2\exp(C_0(\beta_n;n)T)\\ &\quad+\frac{6C_1(\beta_n)}{C_0(\beta_n;n)}T\exp(C_0(\beta_n;n)T)\big(W_2^2(\mu,\hat{\mu})+2C_2\mathbb{E}[h(N^{\#}(\boldsymbol{U}_n(\boldsymbol{x},0);\delta),\ell)]\\ &\quad\cdot(\Theta_6^{\frac{1}{3}}+\hat{\Theta}_6^{\frac{1}{3}})\big)\end{aligned} \tag{D.23}$$

where $C_i, i=0,\dots,2$ are constants defined in Theorem (2.2). Taking the expectation of Eq. (D.20) and plugging in the inequality (D.23) for $\mathbb{E}[\tilde{W}_{2,\delta}^{2,\mathrm{e}}(U_n,\hat{U}_n)]$, the inequality (D.8) is proved. □

Appendix E. Proof of Theorem 3.1

Given a parameterized multivariate normal distribution $f_{\boldsymbol{x}}$, we design an SNN described in Fig. 1 with the ReLU activation and the linear forward propagation. The probability density function of the output of this SNN, denoted by $\hat{f}_{\boldsymbol{x}}$, can approximate $f_{\boldsymbol{x}}$ in the W_2 distance sense. Given a real number $0<c<\epsilon_0$, we choose $\Delta x>0$ such that:

$$W_2^2(f_{\boldsymbol{x}},f_{\tilde{\boldsymbol{x}}})<c,\ \forall \boldsymbol{x},\tilde{\boldsymbol{x}}\in D,\ \|\boldsymbol{x}-\tilde{\boldsymbol{x}}\|<\sqrt{d}\Delta x. \tag{E.1}$$

We consider a uniform equidistance grid set $X := \{\boldsymbol{x}_i\}_{i=1}^K, \boldsymbol{x}_i=(x_i^1,\dots,x_i^d)$ such that the distance between two adjacent points is Δx, $D\subseteq \cup_{i=1}^K\otimes_{j=1}^d[x_i^j,x_i^j+\Delta\boldsymbol{x})$, and $\otimes_{j=1}^d[x_{i_1}^j,x_{i_1}^j+\Delta\boldsymbol{x})\cap\otimes_{j=1}^d[x_{i_2}^j,x_{i_2}^j+\Delta\boldsymbol{x})=\emptyset$ if $i_1\ne i_2$. Therefore, $\forall\boldsymbol{x}=(x^1,\dots,x^d)\in D$, there exists $\boldsymbol{x}_i\in X$ such that $|\boldsymbol{x}-\boldsymbol{x}_i|<\sqrt{d}\Delta x$. Let $0<\epsilon<\frac{1}{2}$ be a small positive number. We set $4dK$ neurons in the first layer, grouped into dK groups. When inputting $\boldsymbol{x}$ into the SNN, the outputs of the four neurons in the $(i,j), i=1,\dots,K, j=1,\dots,d$ group are:

$$\begin{aligned}&n_{i,j,1}^1=\mathrm{ReLU}\Big(\epsilon^{-1}(x^j-x_i^j-\Delta x)\Big),\ n_{i,j,2}^1=\mathrm{ReLU}\Big(\epsilon^{-1}(x^j-x_i^j-\Delta x+\epsilon)\Big),\\ &n_{i,j,3}^1=\mathrm{ReLU}\Big(\epsilon^{-1}\Big(x^j-x_i^j-\epsilon\Big)\Big),\ n_{i,j,4}^1=\mathrm{ReLU}\Big(\epsilon^{-1}\Big(x^j-x_i^j\Big)\Big).\end{aligned} \tag{E.2}$$

The second hidden layer contain dK neurons labeled by $(i,j), i=1,\dots,K, j=1,\dots,d$. We set weights between the first layer and the second layer such that the input of the (i,j) neuron in the second layer is:

$$n_{i,j}^{2,\mathrm{in}} := n_{i,j,2}^1-n_{i,j,1}^1-\Big(n_{i,j,4}^1-n_{i,j,3}^1\Big). \tag{E.3}$$

It is easy to verify that $n_{i,j}^{2,\mathrm{in}}\in[0,1]$. Furthermore, when $x^j\in[x_i^j+\epsilon,x_i^j+\Delta x]$, then $n_{i,j}^{2,\mathrm{in}}=1, j=1,\dots,d$. If $x^j\le x_i^j$ or $x^j\ge x_i^j+\Delta x$, then $n_{i,j}^{2,\mathrm{in}}=0$. The output of the (i,j) neuron in the second hidden layer is designed as

$$n_{i,j}^2=\mathrm{ReLU}\Big(\epsilon^{-1}\Big(n_{i,j}^{2,\mathrm{in}}-1+\epsilon\Big)\Big),\ i=1,\dots,K,\ j=1,\dots d. \tag{E.4}$$

The third layer contains K neurons, and the output of each neuron in the third hidden layer is:

$$n_i^3=\mathrm{ReLU}\left(\sum_{j=1}^d\epsilon^{-1}\Big(n_{i,j}^2-1+\frac{\epsilon}{d}\Big)\right), i=1,\dots,K. \tag{E.5}$$

Thus, $n_i^3\in[0,1]$ and $n_i^3=1$ when $\boldsymbol{x}\in\otimes_j[x_{j,i}+\epsilon,x_{j,i}+\Delta x-\epsilon]$. If there is a $j=1,\dots,d$ such that $x^j<x_i^j$ or $x^j>x_i^j+\Delta x$, then $n_i^3=0$. For $\boldsymbol{x}\in D$, there exists at most one i such that $n_i^3\ne0$. We denote $D(\epsilon)=\{x\in D:\exists!i,n_i^3=1\}$. It is easy to check that as $\epsilon\to0$, $D(\epsilon)\to D$.

We set $d'K$ neurons in the fourth layer. Each neuron is labeled with $(i,j), i=1,\dots,K, k=1,\dots,d'$. The input and output of the (i,j) neuron in the fourth layer are identical (i.e., each neuron in the fourth layer outputs its input):

$$n_i^3(\omega_{i,k}^4+(A_i^{-1}\boldsymbol{b}_i)_k),\ i=1,\dots,K,\ k=1,\dots,d'. \tag{E.6}$$

Here, $\omega_{i,k} \sim \mathcal{N}(0,1)$ are independent random variables, while $\boldsymbol{b}_i$ and A_i are the mean and covariance matrix of $f_{\boldsymbol{x}_i}(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{b}_i, A_i)$, respectively. $(A_i^{-1}\boldsymbol{b}_i)_k$ refers to the kth component of the vector $A_i^{-1}\boldsymbol{b}_i$. The output of the fifth layer is d' dimensional:

$$\sum_{i=1}^{K} n_i^3 (A_i \boldsymbol{\omega}_i^4 + \boldsymbol{b}_i), \tag{E.7}$$

where $\boldsymbol{\omega}_i := (\omega_{i,1}, \ldots, \omega_{i,d'})$. Because for any $\boldsymbol{x} \in D$, there exists at most one i such that $n_i^3 \neq 0$, for any $\boldsymbol{x} \in D$, we have:

$$\begin{aligned} \sup_{\hat{\boldsymbol{y}} \sim f_{\boldsymbol{x}}} \mathbb{E}[\|\boldsymbol{y}\|^2] &\leq \sup_i \mathbb{E}\left[\|A_i \boldsymbol{\omega}_i^4 + \boldsymbol{b}_i\|^2\right] \\ &\leq \sup_i (\|A_i^T A_i\|_F^2 + \|\boldsymbol{b}_i\|_2^2. \end{aligned} \tag{E.8}$$

For each $\boldsymbol{x} \in D(\epsilon)$, if $n_i^3 = 1$ i.e., $\|\boldsymbol{x} - \boldsymbol{x}_i\| < \Delta x$, then $\hat{f}_{\boldsymbol{x}}$ is the probability density function of a multivariate normal distribution, and $\hat{f}_{\boldsymbol{x}}$ is identical to $f_{\boldsymbol{x}_i}$. Therefore, we have

$$\begin{aligned} &\int_D W_2^2(f_{\boldsymbol{x}}, \hat{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) < \int_{D(\epsilon)} W_2^2(f_{\boldsymbol{x}}, \hat{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) \\ &\quad + 2\gamma(D - D(\epsilon)) \cdot \left(\sup_{\boldsymbol{y} \sim f_{\boldsymbol{x}}} \mathbb{E}[\|\boldsymbol{y}\|^2] + \sup_{\hat{\boldsymbol{y}} \sim f_{\boldsymbol{x}}} \mathbb{E}[\|\hat{\boldsymbol{y}}\|^2] \right) \\ &\leq c\gamma(D) + \gamma(D - D(\epsilon)) 2 \left(\sup_{\hat{\boldsymbol{y}} \sim f_{\boldsymbol{x}}} \mathbb{E}[\|\boldsymbol{y}\|^2] + \sup_i \left(\|A_i^T A_i\|_F^2 + \|\boldsymbol{b}_i\|_2^2 \right) \right) \\ &\leq c + 4\gamma(D - D(\epsilon)) Y. \end{aligned} \tag{E.9}$$

Choosing $c < \epsilon_0$ and a small ϵ such that $\gamma(D - D(\epsilon)) \leq \frac{\epsilon_0 - c}{4Y}$, we have proved Theorem 3.1.

Appendix F. Proof of Corollary 3.1

The proof of Corollary 3.1 is based on the proof of Theorem 3.1. Given any positive number $c > 0$, we can find a $\Delta x > 0$ such that:

$$W_2^2(f_{\boldsymbol{x}}, f_{\tilde{\boldsymbol{x}}}) < c, \ \forall \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| < \sqrt{d}\Delta x, \ \boldsymbol{x}, \tilde{\boldsymbol{x}} \in D. \tag{F.1}$$

We establish an equidistance grid point set $X := \{\boldsymbol{x}_i\}_{i=1}^K$ on D such that the distance between two adjacent points is Δx, $D \subseteq \cup_{i=1}^K \otimes_{j=1}^d [x_i^j, x_i^j + \Delta \boldsymbol{x})$ and $\otimes_{j=1}^d [x_{i_1}^j, x_{i_1}^j + \Delta \boldsymbol{x}) \cap \otimes_{j=1}^d [x_{i_2}^j, x_{i_2}^j + \Delta \boldsymbol{x}) = \emptyset$ if $i_1 \neq i_2$. Thus, $\forall \boldsymbol{x} \in D$, there exists $\boldsymbol{x}_i \in X$ such that $|\boldsymbol{x} - \boldsymbol{x}_i| \leq \sqrt{d}\Delta x$. Denote $\Phi(x)$ to be the cumulative distribution function of a standard normal random variable. Suppose $-M = h_{i,0} < h_{i,1} < \ldots < h_{i,s} = M$ such that:

$$\begin{aligned} &\Phi(h_{i,r+1}) - \Phi(h_{i,r}) = p_{i,r+1}, r = 1, \ldots, s-2, \ \Phi(h_{i,1}) = p_{i,1}, \\ &\Phi(h_{i,s-1}) = 1 - p_{i,s}, \end{aligned} \tag{F.2}$$

where $p_{i,r} := p_r(\boldsymbol{x}_i)$. We design an SNN of the form in Fig. 1 that satisfies Eq. (3.12). We set the first three hidden layers of this SNN to be the same as the first three layers of the SNN designed in the proof of Theorem 3.1 in Appendix E. We also refer to the outputs of the third hidden layer as $n_i^3, i = 1, \ldots, K$. For $\boldsymbol{x}_i \in X$, we denote $\boldsymbol{x}_i = (x_i^1, \ldots, x_i^d)$. From Appendix E, $n_i^3 \in [0,1]$, $n_i^3 = 1$ when $\boldsymbol{x} = (x^1, \ldots, x^d) \in \otimes_{j=1}^d [x_i^j + \epsilon, x_i^j + \Delta x - \epsilon]$, and $n_i^3 = 0$ when $\boldsymbol{x} \in D - \otimes_{j=1}^d [x_i^j, x_i^j + \Delta x]$. $\epsilon > 0$ is a small number to be determined. If there is a $j = 1, \ldots, d$ such that $x^j < x_i^j$ or $x^j > x_i^j + \Delta x$, then $n_i^3 = 0$. For $\boldsymbol{x} \in D$, there is at most one i such that $n_i^3 = 1$.

The fourth layer contains $K(s+1)$ groups of neurons with each group containing 2 neurons. The outputs of the $(i,r), i = 1, \ldots, K, r = 0, \ldots, s$ group are:

$$n_{i,r}^4 = \mathrm{ReLU}(\tilde{w}_i^3 n_i^3 - h_{i,r} - M_0), \tilde{n}_{i,r}^4 = \mathrm{ReLU}(\tilde{w}_i^3 n_i^3 - h_{i,r} - \epsilon_0 - M_0). \tag{F.3}$$

$\tilde{w}_i^3 \sim \mathcal{N}(M_0, 1)$ are independent random variables. $M_0 > |h_{i,r}|, \forall i, r$ is a large number and $\epsilon_0 > 0$ is a small number satisfying $\epsilon_0 < \min_{i,r} \frac{h_{i,r+1} - h_{i,r}}{2}$. Both M_0 and ϵ_0 are to be determined.

The fifth layer contains sK neurons and the output of the (i,r) neuron is:

$$n_{i,r+1}^5 = \mathrm{ReLU}\left(\epsilon_0^{-1}\left(n_{i,r}^4 - \tilde{n}_{i,r}^4 - \left(n_{i,r+1}^4 - \tilde{n}_{i,r+1}^4\right)\right)\right), \ r = 0, \ldots, s-1. \tag{F.4}$$

Therefore, when $n_i^3 = 1$, for $r = 0, \ldots, s-1$, we have:

$$\begin{aligned} &n_{i,r+1}^5 = 1, \omega_i^3 \in [M_0 + h_{i,r} + \epsilon_0, M_0 + h_{i,r+1}], \\ &n_{i,r+1}^5 = 0, \omega_i^3 < M_0 + h_{i,r} \text{ or } \omega_i^3 > M_0 + h_{i,r+1} + \epsilon_0, \\ &n_{i,r+1}^5 \in (0,1), \text{otherwise}; \end{aligned} \tag{F.5}$$

Furthermore, we can see that for any $n_i^3 \in [0,1]$

$$0 \leq \sum_{r=1}^{s} n_{i,r}^5 \leq 1, \tag{F.6}$$

and there exist at most two non-zero $n_{i,s}^5, n_{i+1,s}^5 > 0$ in $(n_{i,1}^5, \ldots, n_{i,s}^5)$.

For any $\epsilon_1 > 0$, there exist a small $\epsilon_0 > 0$ and a large $M_0 > 0$ such that when $n_i^3 = 1$:

$$0 \leq p_{i,r} - \hat{p}_{i,r} \leq \frac{\epsilon_1}{s}, \ \hat{p}_{i,r} := p(n_{i,r}^5 = 1), \ i = 1, \ldots, K. \tag{F.7}$$

The sixth hidden layer contains sKd' neurons, each of whose input and output are both

$$n_{i,r,k}^6 = n_{i,r}^5 (w_{i,r,k} + (A_{i,r}^{-1}\boldsymbol{b}_{i,r})_k), i = 1, \ldots, K, r = 1, \ldots, s, k = 1, \ldots, d'. \tag{F.8}$$

Here, $w_{i,r,k} \sim \mathcal{N}(0,1)$ are independent random variables. $\boldsymbol{b}_{i,s}$ and $A_{i,s}$, respectively, are the mean vector and covariance matrix such that

$$f_{\boldsymbol{x}_i}(\boldsymbol{y}) = \sum_{r=1}^{s} p_r(\boldsymbol{x}_i) \mathcal{N}\left(\boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r}\right). \tag{F.9}$$

The seventh layer contains d' neurons whose output is:

$$\sum_{i=1}^{K} \sum_{r=1}^{s} n_{i,r}^5 (A_{i,r} \boldsymbol{w}_{i,r} + \boldsymbol{b}_{i,r}), \tag{F.10}$$

where $\boldsymbol{w}_{i,r} := (w_{i,r,1}, \ldots, w_{i,r,d'})$. When $\boldsymbol{x} \in D(\epsilon) := \{x \in D : \exists! i, n_i^3 = 1\}$, if $n_i^3 = 1$, then the probability density function of the output of the seventh layer can be written as:

$$\sum_{r=1}^{s} \hat{p}_{i,r} \mathcal{N}\left(\boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r}\right) + p_i(\boldsymbol{y}), \ \int_{\mathbb{R}^{d'}} p_i(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} := 1 - \sum_{r=1}^{s} \hat{p}_{i,r} \leq \epsilon_1, \ p_i(\boldsymbol{y}) \geq 0. \tag{F.11}$$

Furthermore, since there exist at most two non-zero consecutive $n_{i,s}^5, n_{i+1,s}^5 > 0$, we have:

$$\begin{aligned} \int_{\mathbb{R}^{d'}} \|\boldsymbol{y}\|^2 p_i(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} &\leq \int_{\mathbb{R}^{d'}} p_i(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} \cdot \left(2 \max_{i,r} \mathbb{E}_{\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r})}[\|\boldsymbol{y}\|^2] \right. \\ &\quad \left. + 2 \max_{i,r} \mathbb{E}_{\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r})}[\|\boldsymbol{y}\|^2] \right) \\ &= (1 - \sum_{r=1}^{s} \hat{p}_{i,r}) 4 \max_r \left(\|\boldsymbol{b}_{i,r}\|^2 + \|A_{i,r}^T A_{i,r}\|_F^2 \right) \\ &\leq 4\epsilon_1 \max_r \left(\|\boldsymbol{b}_{i,r}\|^2 + \|A_{i,r}^T A_{i,r}\|_F^2 \right). \end{aligned} \tag{F.12}$$

Applying Lemma 3.1, we have

$$\begin{aligned} W_2^2(\hat{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}_i}) &\leq 2 \max_r \frac{\epsilon_1}{s} \cdot s \left(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2 \right) \\ &\quad + 4\epsilon_1 \max_r \left(\|\boldsymbol{b}_{i,r}\|^2 + \|A_{i,r}^T A_{i,r}\|_F^2 \right), \end{aligned} \tag{F.13}$$

when $\boldsymbol{x} \in D^i(\epsilon) := \{\boldsymbol{x} \in D(\epsilon) | \|\boldsymbol{x}_i - \boldsymbol{x}\| \le \|\boldsymbol{x}_j - \boldsymbol{x}\|, \forall j \ne i\}$. Finally, we have:

$$\begin{aligned}
&\int_D W_2^2(f_{\boldsymbol{x}}, \hat{f}_{\boldsymbol{x}})\gamma(\mathrm{d}\boldsymbol{x}) \\
\le &\int_{D(\epsilon)} W_2^2(f_{\boldsymbol{x}}, \hat{f}_{\boldsymbol{x}})\gamma(\mathrm{d}\boldsymbol{x}) \\
&+ 2(1-\gamma(D(\epsilon)))\Big(\mathbb{E}[\|\boldsymbol{y}\|^2] + 4\sup_{i,r}\left(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2\right)\Big) \\
\le &\, 2\sum_i \int_{D^i(\epsilon)} W_2^2(\hat{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}_i})\gamma(\mathrm{d}\boldsymbol{x}) + 2\sum_i \int_{D^i(\epsilon)} c\gamma(\mathrm{d}\boldsymbol{x}) \\
&+ 2(1-\gamma(D(\epsilon)))\Big(\mathbb{E}[\|\boldsymbol{y}\|^2] + 4\sup_{i,r}(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2)\Big) \\
\le &\, 2\sup_{i,r}\Big(6\epsilon_1\left(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2\right)\Big) + 2c \\
&+ 2(1-\gamma(D(\epsilon)))\Big(\mathbb{E}[\|\boldsymbol{y}\|^2] + 4\sup_{i,r}(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2)\Big).
\end{aligned} \tag{F.14}$$

Note that $\sup_{i,r}(\|A_{i,r}^T A_{i,r}\|_F^2 + \|\boldsymbol{b}_{i,r}\|^2)$ is uniformly bounded from the assumption Eq. (3.11). Letting $c, \epsilon, \epsilon_1 \to 0^+$ in Eq. (F.14), we have proved the inequality (3.12).

Appendix G. Proof of Theorem 3.2

First, we define an auxiliary function with an additional parameter σ:

$$f_{\sigma^2}(\boldsymbol{y}) := \int_{\mathbb{R}^{d'}} f(\boldsymbol{y}')\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}'. \tag{G.1}$$

Since f is uniformly continuous and uniformly bounded, $\forall \epsilon_0 > 0$, there exist a small $\delta > 0$ and a small $\sigma_0 > 0$ such that for any $\sigma < \sigma_0$, we have i) $|f(\boldsymbol{y}) - f(\tilde{\boldsymbol{y}})| < \epsilon_0, \forall \|\tilde{\boldsymbol{y}} - \boldsymbol{y}\| < \delta$ and ii)

$$\int_{B(0,\delta)} \mathcal{N}(\boldsymbol{y}; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y} > 1 - \epsilon_0. \tag{G.2}$$

Therefore, we conclude that $\lim_{\sigma\to 0} f_{\sigma^2}(\boldsymbol{y}) = f(\boldsymbol{y})$ uniformly on $\mathbb{R}^{d'}$.

Letting $\{\boldsymbol{y}_j\}_{j=1}^{(n_0+1)^{d'}}$ and $\{w_j\}_{j=1}^{(n_0+1)^{d'}}$ be the multidimensional Hermite collocation points and weights on $\mathbb{R}^{d'}$ described in Shen et al. (2011), we have

$$\begin{aligned}
&\int_{\mathbb{R}^{d'}} \mathcal{I}_{n_0} f(\boldsymbol{y}') \cdot \mathcal{I}_{n_0}\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}' \\
= &\sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_i)\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i; \sigma^2 I_{d'\times d'})w_i.
\end{aligned} \tag{G.3}$$

Here, $\mathcal{I}_{n_0}$ is the interpolation operator such that

$$f(\boldsymbol{y}_j) = \mathcal{I}_{n_0} f(\boldsymbol{y}_j) \in P_{n_0},\ j = 1, \dots, (n_0+1)^{d'}, \tag{G.4}$$

where P_{n_0} is the space spanned by the generalized Hermite functions $\hat{\mathcal{H}}_{\boldsymbol{n}}(\boldsymbol{y})$ such that $|\boldsymbol{n}|_\infty \le n_0$ and

$$\hat{\mathcal{H}}_{\boldsymbol{n}}(\boldsymbol{y}) := \prod_{i=1}^{d'} \hat{\mathcal{H}}_{n_i}(y_i),\ \boldsymbol{n} = (n_1, \dots, n_{d'}),\ \boldsymbol{y} = (y_1, \dots, y_{d'}) \tag{G.5}$$

is the multidimensional generalized Hermite function defined in Shen et al. (2011) ($\hat{\mathcal{H}}_{n_i}$ is the 1D generalized Hermite function of order n_i). We denote

$$\begin{aligned}
f_{\sigma^2,n_0}(\boldsymbol{y}) &:= \sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_i)\cdot\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i; \sigma^2 I_{d'\times d'})w_i \\
&= \int_{\mathbb{R}^{d'}} \mathcal{I}_{n_0} f(\boldsymbol{y}')\cdot\mathcal{I}_{n_0}\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}',
\end{aligned} \tag{G.6}$$

where f_{σ^2,n_0} is nonnegative because the collocation weights $w_j > 0$. Furthermore,

$$\begin{aligned}
&\Big|\int_{\mathbb{R}^{d'}} f(\boldsymbol{y}')\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'}) - \mathcal{I}_{n_0} f(\boldsymbol{y}')\mathcal{I}_{n_0}\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}'\Big| \\
\le &\, \|f - \mathcal{I}_{n_0} f\|_{L^2}\cdot\|\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\|_{L^2} \\
&+ \|\mathcal{I}_{n_0} f\|_{L^2}\cdot\|\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'}) - \mathcal{I}_{n_0}\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\|_{L^2}.
\end{aligned} \tag{G.7}$$

Using (Shen et al., 2011, Theorem 7.18, Theorem 8.6), we have:

$$\begin{aligned}
\|f - \mathcal{I}_{n_0} f\| &\le \|f - \mathcal{I}_{n_0}^1 f\|_{L^2} + \|\mathcal{I}_{n_0}^2\circ\dots\circ\mathcal{I}_{n_0}^{d'} f - f\|_{L^2} \\
&\quad + \|(\mathcal{I}_{n_0}^1 - \mathbb{I})\circ(\mathcal{I}_{n_0}^2\circ\dots\circ\mathcal{I}_{n_0}^{d'} f - f)\|_{L^2}, \\
&\le Cn_0^{-\frac{1}{3}}\|\partial_{y_1} f\|_{L^2} + \|\mathcal{I}_{n_0}^2\circ\dots\circ\mathcal{I}_{n_0}^{d'} f - f\|_{L^2} \\
&\quad + \|\mathcal{I}_{n_0}^2\circ\dots\circ(\mathcal{I}_{n_0}^1 - \mathbb{I})(\mathcal{I}_{n_0}^{d'} f - f)\|_{L^2}, \\
&\le Cn_0^{-\frac{1}{3}}\|\partial_{y_1} f\|_{L^2} + \|\mathcal{I}_{n_0}^2\circ\dots\circ\mathcal{I}_{n_0}^{d'} f - f\|_{L^2} \\
&\quad + Cn_0^{-\frac{1}{3}}\|(\mathcal{I}_{n_0}^2\circ\dots\circ\mathcal{I}_{n_0}^{d'}\partial_{y_1} f - \partial_{y_1} f\|_{L^2} \\
&\le \dots \\
&\le Cn_0^{-\frac{1}{3}}|f|_{\mathrm{mix}}
\end{aligned} \tag{G.8}$$

Here, C is a constant and n_0 is taken to be large enough such that $Cn_0^{-\frac{1}{3}} < 1$. $\mathbb{I}$ is the identity operator and $\mathcal{I}_{n_0}^i, i = 1, \dots, d'$ is the projection operator in the ith direction, i.e., if we denote $X_{n_0} := \{y_i\}_{i=0}^{n_0}$ to be the 1D Hermite collocation points, then

$$\mathcal{I}_{n_0}^i f(\boldsymbol{y}) = f(\boldsymbol{y}),\ \forall \boldsymbol{y} = (y_1, \dots, y_{d'}) \text{ if } y_i \in X_{n_0}. \tag{G.9}$$

Similarly, for any fixed $\boldsymbol{y} \in \mathbb{R}^{d'}$:

$$\begin{aligned}
&\|\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'}) - \mathcal{I}_{n_0}\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\|_{L^2} \\
&\le \sum_{|\boldsymbol{n}|_0 \le n_0} Cn_0^{-\frac{1}{3}}\|\partial_{\boldsymbol{n}}\mathcal{N}\|_{L^2}.
\end{aligned} \tag{G.10}$$

Combining Eqs. (G.3), (G.7), (G.8) and (G.10), for every $\boldsymbol{y}$ and every $\sigma > 0$, as $n_0 \to \infty$,

$$\begin{aligned}
&\Big|\sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_j)\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_j; \sigma^2 I_{d'\times d'})w_j - \int_{\mathbb{R}^{d'}} f(\boldsymbol{y}')\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}'\Big| \\
&\le Cn_0^{-\frac{1}{3}}\Big(|f|_{\mathrm{mix}}\|\mathcal{N}(\boldsymbol{y}, \sigma^2 I_{d'\times d'})\|_{L^2} + \left(\|f\|_{L^2} + Cn_0^{-\frac{1}{3}}|f|_{\mathrm{mix}}\right)| \\
&\cdot |\mathcal{N}(\boldsymbol{y}, \sigma^2 I_{d'\times d'})|_{\mathrm{mix}}\Big),
\end{aligned} \tag{G.11}$$

which implies that for any fixed $\sigma > 0$,

$$\begin{aligned}
f_{\sigma^2,n_0}(\boldsymbol{y}) &= \sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_i)\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i; \sigma^2 I_{d'\times d'})w_i \\
&\to \int_{\mathbb{R}^{d'}} f(\boldsymbol{y}')\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}'; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y}'
\end{aligned} \tag{G.12}$$

uniformly in $\boldsymbol{y} \in \mathbb{R}^{d'}$ as $n_0 \to \infty$.

Additionally, we have:

$$\begin{aligned}
&\int_{\mathbb{R}^{d'}} \|\boldsymbol{y}\|^2 \sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_i)w_i\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i; \sigma^2 I_{d'\times d'})\mathrm{d}\boldsymbol{y} \\
&= \sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_i)(|\boldsymbol{y}_i|^2 + d'\sigma^2)w_i \\
&= \sum_{i=1}^{(n_0+1)^{d'}} f(\boldsymbol{y}_j)(|\boldsymbol{y}_i|^2 + d'\sigma^2)^2\frac{1}{|\boldsymbol{y}_i|^2 + d'\sigma^2}w_i \\
&= \int_{\mathbb{R}^{d'}} \mathcal{I}_{n_0}\left(f(\boldsymbol{y})(\|\boldsymbol{y}\|^2 + d'\sigma^2)^2\right)\cdot\mathcal{I}_{n_0}\left(\frac{1}{(\|\boldsymbol{y}\|^2 + d'\sigma^2)}\right)\mathrm{d}\boldsymbol{y} \\
&\le \int_{\mathbb{R}^{d'}} f(\boldsymbol{y})(\|\boldsymbol{y}\|^2 + d'\sigma^2)^2\cdot\frac{1}{(\|\boldsymbol{y}\|^2 + d'\sigma^2)}\mathrm{d}\boldsymbol{y} \\
&\quad + \left\|(\mathcal{I}_{n_0} - \mathbb{I})\left(f(\boldsymbol{y})(\|\boldsymbol{y}\|^2 + d'\sigma^2)^2\right)\right\|_{L^2}\cdot\left\|\frac{1}{\|\boldsymbol{y}\|^2 + d'\sigma^2}\right\|_{L^2}
\end{aligned}$$

$$
\begin{aligned}
&+\|\mathcal{I}_{n_0}\big(f(\boldsymbol{y})(\|\boldsymbol{y}\|^2+d'\sigma^2)^2\big)\|_{L^2}\cdot\|(\mathbb{I}-\mathcal{I}_{n_0})\big(\frac{1}{\|\boldsymbol{y}\|^2+d'\sigma^2}\big)\|_{L^2}\\
&\le\Big(\sum_{i,j=1}^{d'}2Cn_0^{-\frac{1}{3}}|f(\boldsymbol{y})y_i^2y_j^2|_{\mathrm{mix}}+2Cn_0^{-\frac{1}{3}}d'\sigma^2\sum_{i=1}^{d'}|f(\boldsymbol{y})y_i^2|_{\mathrm{mix}}\\
&\quad+Cn_0^{-\frac{1}{3}}(d')^2\sigma^4|f|_{\mathrm{mix}}\Big)\cdot\sigma^{-\frac{1}{2}}C_1(d')\\
&\quad+\|\mathcal{I}_{n_0}\big(f(\boldsymbol{y})(\boldsymbol{y}+d'\sigma^2)^2\big)\|_{L^2}\cdot Cn_0^{-\frac{1}{3}}\Big|\frac{1}{\|\boldsymbol{y}\|^2+d'\sigma^2}\Big|_{\mathrm{mix}}+\mathbb{E}[\|\boldsymbol{y}\|^2]+d'\sigma^2,
\end{aligned}
\tag{G.13}
$$

where $C_1(d')$ is another constant depending on the dimensionality d'. Since

$$
\begin{aligned}
&\|\mathcal{I}_{n_0}\big(f(\boldsymbol{y})(\|\boldsymbol{y}\|^2+d'\sigma^2)^2\big)-f(\boldsymbol{y})(\|\boldsymbol{y}\|^2+d'\sigma^2)^2\|_{L^2}\\
&\le\sum_{i,j=1}^{d'}2Cn_0^{-\frac{1}{3}}|f(\boldsymbol{y})y_i^2y_j^2|_{\mathrm{mix}}+2Cn_0^{-\frac{1}{3}}d'\sigma^2\sum_{i=1}^{d'}|f(\boldsymbol{y})y_i^2|_{\mathrm{mix}}\\
&\quad+Cn_0^{-\frac{1}{3}}(d')^2\sigma^4|f|_{\mathrm{mix}}\le\infty,
\end{aligned}
\tag{G.14}
$$

we conclude that

$$
\int_{\mathbb{R}^{d'}}\|\boldsymbol{y}\|^2\sum_{i=1}^{(n_0+1)^{d'}}f(\boldsymbol{y}_i)w_i\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i;\sigma^2)\mathrm{d}\boldsymbol{y}<\infty.
\tag{G.15}
$$

Furthermore, from the inequality (G.14), $\|\mathcal{I}_{n_0}\big(f(\boldsymbol{y})(\|\boldsymbol{y}\|^2+d'\sigma^2)^2\big)\|_{L^2}\to\|f(\boldsymbol{y})(\|\boldsymbol{y}\|^2+d'\sigma^2)^2\|_{L^2}$ as $n_0\to\infty$. We denote

$$
\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y}):=\frac{1}{\sum_{j=1}^{(n_0+1)^{d'}}f(\boldsymbol{y}_j)w_j}f_{\sigma^2,n_0}(\boldsymbol{y}).
\tag{G.16}
$$

Therefore, $\int_{\mathbb{R}^{d'}}\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})\mathrm{d}\boldsymbol{y}=1$. Note that:

$$
\begin{aligned}
&\Big|\int_{\mathbb{R}^{d'}}f(\boldsymbol{y})\mathrm{d}\boldsymbol{y}-\sum_{i=1}^{(n_0+1)^{d'}}f(\boldsymbol{y}_i)w_i\Big|=\int_{\mathbb{R}^{d'}}f-\mathcal{I}_{n_0}\sqrt{f}\cdot\mathcal{I}_{n_0}\sqrt{f}\mathrm{d}\boldsymbol{y}\\
&\le\|\sqrt{f}\|_{L^2}Cn_0^{-\frac{1}{3}}|\sqrt{f}|_{\mathrm{mix}}+\|\mathcal{I}_{n_0}\sqrt{f}\|_{L^2}Cn_0^{-\frac{1}{3}}|\sqrt{f}|_{\mathrm{mix}},\\
&\le Cn_0^{-\frac{1}{3}}|\sqrt{f}|_{\mathrm{mix}}(2\|\sqrt{f}\|_{L^2}+Cn_0^{-\frac{1}{3}}|\sqrt{f}|_{\mathrm{mix}}).
\end{aligned}
\tag{G.17}
$$

Therefore, we can write

$$
\sum_{i=1}^{n_0^{d'}}f(\boldsymbol{y}_i)w_i:=1+n_0^{-\frac{1}{3}}c(\sqrt{f})\le\infty,
\tag{G.18}
$$

where $c(\sqrt{f})$ is a constant depending on $|\sqrt{f}|_{\mathrm{mix}}$ ($\|\sqrt{f}\|_{L^2}=1$). We assume n_0 is large enough such that $n_0^{-\frac{1}{3}}c(\sqrt{f})<\frac{1}{2}$. From the inequality (G.13) and the definition of $\tilde{f}_{\sigma^2,n_0}$ in Eq. (G.16), we can bound $\mathbb{E}_{\boldsymbol{y}\sim\tilde{f}_{\sigma^2,n_0}}[\|\boldsymbol{y}\|^2]$ by

$$
\begin{aligned}
\mathbb{E}_{\boldsymbol{y}\sim\tilde{f}_{\sigma^2,n_0}}[\|\boldsymbol{y}\|^2]&\le\frac{1}{1-|c(f)|n_0^{-\frac{1}{3}}}\Big[\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+d'\sigma^2\\
&\quad+n_0^{-\frac{1}{3}}(C_2(\sigma;f)+\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+1)\Big]\\
&\le\Big[\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+n_0^{-\frac{1}{3}}(C_2(\sigma;f)+\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+1)+d'\sigma^2\Big]\\
&\quad+2|c(\sqrt{f})|n_0^{-\frac{1}{3}}\Big[\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+n_0^{-\frac{1}{3}}(C_2(\sigma;f)\\
&\quad+\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+1)+d'\sigma^2\Big]\\
&=\mathbb{E}[\|\boldsymbol{y}\|^2]+d'\sigma^2+n_0^{-\frac{1}{3}}C_3(\sigma;f),
\end{aligned}
\tag{G.19}
$$

where $C_2(\sigma;f)$ is a constant that depends on f and σ, and $C_3(\sigma;f)$ is another constant depending on f, $\sqrt{f}$ and σ.

Consider the special coupling measure

$$
\begin{aligned}
&\pi(f,\tilde{f}_{\sigma^2,n_0})(\boldsymbol{y},\hat{\boldsymbol{y}}):=\min\big(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})\big)\delta(\boldsymbol{y}-\hat{\boldsymbol{y}})\\
&\quad+\frac{1}{A}\big(f(\boldsymbol{y})-\min(f,\tilde{f}_{\sigma^2,n_0})(\boldsymbol{y})\big)\cdot\big(\tilde{f}_{\sigma^2,n_0}(\hat{\boldsymbol{y}})-\min(f,\tilde{f}_{\sigma^2,n_0})(\hat{\boldsymbol{y}})\big),\\
&\text{if}\int_{\mathbb{R}^{d'}}\min(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y}))\mathrm{d}\boldsymbol{y}<1,\\
&\pi(f,\tilde{f}_{\sigma^2,n_0})(\boldsymbol{y},\hat{\boldsymbol{y}}):=f(\boldsymbol{y})\delta(\boldsymbol{y}-\hat{\boldsymbol{y}}),\ \text{if}\int_{\mathbb{R}^{d'}}\min(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y}))\mathrm{d}\boldsymbol{y}=1,
\end{aligned}
\tag{G.20}
$$

where $A:=\int_{\mathbb{R}^{d'}}\min\big(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})\big)\mathrm{d}\boldsymbol{y}$ and δ is the Dirac delta measure. The marginal probability densities of $\pi(f,\tilde{f}_{\sigma^2,n_0})$ are $f(\boldsymbol{y})$ and $\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})$, respectively. Furthermore, we have:

$$
\begin{aligned}
&\mathbb{E}_{(\boldsymbol{y},\hat{\boldsymbol{y}})\sim\pi(f,\tilde{f}_{\sigma^2,n_0})}[\|\boldsymbol{y}-\hat{\boldsymbol{y}}\|^2]\\
&\le2\int_{\mathbb{R}^{d'}}\|\boldsymbol{y}\|^2(f(\boldsymbol{y})-\min(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})))\mathrm{d}\boldsymbol{y}\\
&\quad+2\int_{\mathbb{R}^{d'}}\|\hat{\boldsymbol{y}}\|^2(\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})-\min(f(\boldsymbol{y}),\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})))\mathrm{d}\boldsymbol{y}\\
&\le4\int_{\mathbb{R}^{d'}}\|\boldsymbol{y}\|^2|f(\boldsymbol{y})-\tilde{f}_{\sigma^2,n_0}(\boldsymbol{y})|\mathrm{d}\boldsymbol{y}.
\end{aligned}
\tag{G.21}
$$

Fixing $\sigma>0$, from Eq. (G.12), $f_{\sigma^2,n_0}\to f_{\sigma^2}$ uniformly as $n_0\to\infty$; furthermore, from the definition of $\tilde{f}_{\sigma^2,n_0}$ in Eq. (G.16), $\tilde{f}_{\sigma^2,n_0}\to f_{\sigma^2,n_0}$ uniformly as $n_0\to\infty$. Finally, since $\lim_{\sigma\to0}f_{\sigma^2}(\boldsymbol{y})=f(\boldsymbol{y})$ uniformly as $\sigma\to0$ for any $\boldsymbol{y}\in\mathbb{R}^{d'}$, we conclude that

$$
\tilde{f}_{\sigma^2,n_0(\sigma)}\to f
\tag{G.22}
$$

uniformly as $\sigma\to0$ and $n_0(\sigma)\to\infty$ in $\mathbb{R}^{d'}$.

Since $\mathbb{E}_{\boldsymbol{y}\sim f}\|\boldsymbol{y}\|^2]<\infty$ and $\mathbb{E}_{\boldsymbol{y}\sim\tilde{f}_{\sigma^2,n_0(\sigma)}}[\|\boldsymbol{y}\|^2]\to\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+d'\sigma^2$ as $n_0(\sigma)\to\infty$ from Eq. (G.19), for any $\epsilon>0$, there exists a measurable set $A\subseteq\mathbb{R}^{d'}$ such that:

1. $|\int_A\|\boldsymbol{y}\|^2f(\boldsymbol{y})\mathrm{d}\boldsymbol{y}-\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]|<\epsilon$
2. we can find a sufficiently small σ and a sufficiently large $n_0(\sigma)$ such that $d'\sigma^2<\epsilon$ and:

$$
\int_A\|\boldsymbol{y}\|^2\cdot|f(\boldsymbol{y})-\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})|\mathrm{d}\boldsymbol{y}\le\epsilon.
\tag{G.23}
$$

Using the inequality (G.19), we have

$$
\begin{aligned}
&\mathbb{E}_{(\boldsymbol{y},\hat{\boldsymbol{y}})\sim\pi(f,\tilde{f}_{\sigma^2,n_0(\sigma)})}[\|\boldsymbol{y}-\hat{\boldsymbol{y}}\|^2]\le4\int_A\|\boldsymbol{y}\|^2|f(\boldsymbol{y})-\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})|\mathrm{d}\boldsymbol{y}\\
&\quad+4\int_{\mathbb{R}^{d'}-A}\|\boldsymbol{y}\|^2f(\boldsymbol{y})\mathrm{d}\boldsymbol{y}+4\int_{\mathbb{R}^{d'}-A}\|\boldsymbol{y}\|^2\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})\mathrm{d}\boldsymbol{y}\\
&\le4\epsilon+4\epsilon+4\Bigg(\mathbb{E}_{\boldsymbol{y}\sim f}[\|\boldsymbol{y}\|^2]+d'\sigma^2+n_0(\sigma)^{-\frac{1}{3}}C_3(\sigma;f)\\
&\quad-\Big(\int_A\|\boldsymbol{y}\|^2f(\boldsymbol{y})\mathrm{d}\boldsymbol{y}-\epsilon\Big)\Bigg)\\
&\le16\epsilon+4d'\sigma^2+4n_0(\sigma)^{-\frac{1}{3}}C_3(\sigma;f)\le24\epsilon,
\end{aligned}
\tag{G.24}
$$

if we take an $n_0(\sigma)$ large enough such that $n_0(\sigma)^{-\frac{1}{3}}C_3(\sigma;f)\le\epsilon$. From Eq. (G.16), we have:

$$
\tilde{f}_{\sigma^2,n_0(\sigma)}(\boldsymbol{y})=\sum_{i=1}^{(n_0(\sigma)+1)^{d'}}\frac{f(\boldsymbol{y}_i)w_i}{\sum_{j=1}^{(n_0(\sigma)+1)^{d'}}f(\boldsymbol{y}_j)w_j}\cdot\mathcal{N}(\boldsymbol{y}-\boldsymbol{y}_i;\sigma^2I_{d'\times d'}),
\tag{G.25}
$$

which is indeed the probability density function of a Gaussian mixture model, and this completes the proof of Theorem 3.2.

Appendix H. The approximation ability of the SNN model in Fig. 1

In this subsection, we analyze the capability of the SNN model to approximate a family of probability density functions for a random variable $\boldsymbol{y}_{\boldsymbol{x}}\sim f_{\boldsymbol{x}},\boldsymbol{y}\in\mathbb{R}^{d'}$ characterized by $\boldsymbol{x}\in D\subseteq\mathbb{R}^d,\boldsymbol{x}\sim\gamma(\cdot)$. We assume the following conditions hold:

1. $f_{\boldsymbol{x}}$ is uniformly continuous such that for any ϵ, there exists a Δx satisfying $W_2^2(f_{\boldsymbol{x}}, f_{\tilde{\boldsymbol{x}}}) \le \epsilon,\ \forall \boldsymbol{x}, \tilde{\boldsymbol{x}} \in D, \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| \le \Delta x$.
2. For any $\boldsymbol{x}$, $f_{\boldsymbol{x}}$ satisfies the conditions in Theorem 3.2.

Let $\epsilon > 0$ be a small positive number. We can find a $\Delta x > 0$ such that $W_2^2(f_{\boldsymbol{x}}, f_{\tilde{\boldsymbol{x}}}) < \frac{\epsilon}{4}$ if $\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\| \le \sqrt{d}\Delta x, \forall \boldsymbol{x}\, \tilde{\boldsymbol{x}} \in D$. We can find an equidistance grid point set $X := \{\boldsymbol{x}_i\}_{i=1}^K$ such that the distance between two adjacent points is $\Delta \boldsymbol{x}$, $D \subseteq \cup_{i=1}^K \otimes_{j=1}^d [x_i^j, x_i^j + \Delta \boldsymbol{x})$, and $\otimes_{j=1}^d [x_{i_1}^j, x_{i_1}^j + \Delta \boldsymbol{x}) \cap \otimes_{j=1}^d [x_{i_2}^j, x_{i_2}^j + \Delta \boldsymbol{x}) = \emptyset$ if $i_1 \ne i_2$. Thus, for any $\boldsymbol{x} \in D$, there exists $\boldsymbol{x}_i \in X$ satisfying $W_2^2(f_{\boldsymbol{x}}, f_{\boldsymbol{x}_i}) < \frac{\epsilon}{4}$. Furthermore, for any $\boldsymbol{x}_i = (x_i^1, \ldots, x_i^d) \in X$, from Theorem 3.2, there exists a probability density function of a Gaussian mixture model:

$$\tilde{f}_{n_{0,i},\sigma_i^2}(\boldsymbol{y}_{\boldsymbol{x}_i}) = \sum_{r=1}^{n_{0,i}} p_{i,r} \mathcal{N}\Big(\boldsymbol{y}_{\boldsymbol{x}_i} - \boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r}\Big),\ \sum_{r=1}^{n_{0,i}} p_{i,r} = 1, \tag{H.1}$$

such that $W_2^2(f_{\boldsymbol{x}_i}, \tilde{f}_{n_{0,i},\sigma_i^2}) < \frac{\epsilon}{4}, i = 1, \ldots, K$. We denote $n_0 := \max_{1 \le i \le K} n_{0,i}$ and

$$\begin{aligned} f_{n_0,\sigma_i^2}(\boldsymbol{y}_{\boldsymbol{x}_i}) = &\sum_{r=1}^{n_{0,i}-1} p_{i,r} \mathcal{N}\Big(\boldsymbol{y}_{\boldsymbol{x}_i} - \boldsymbol{b}_{i,r}, A_{i,r}^T A_{i,r}\Big) \\ &+ \sum_{r=n_{0,i}}^{n_0} \frac{p_{i,n_{0,i}}}{n_0 - n_{0,i} + 1} \mathcal{N}\Big(\boldsymbol{y}_{\boldsymbol{x}_i} - \boldsymbol{b}_{i,n_{0,i}}, A_{i,n_{0,i}}^T A_{i,n_{0,i}}\Big). \end{aligned} \tag{H.2}$$

$W_2^2(f_{\boldsymbol{x}_i}, f_{n_{0,i},\sigma_i^2}) < \frac{\epsilon}{4}, i = 1, \ldots, K$ for $f_{n_{0,i},\sigma_i^2}$ defined in Eq. (H.2) because $W_2^2(f_{\boldsymbol{x}_i}, f_{n_{0,i},\sigma_i^2}) = W_2^2(\tilde{f}_{\boldsymbol{x}_i}, f_{n_{0,i},\sigma_i^2})$.

We define a new continuous random variable $\tilde{\boldsymbol{y}}_{\boldsymbol{x}}$ with a probability distribution $\tilde{f}_{\boldsymbol{x}}, \boldsymbol{x} \in D$ such that:

$$\tilde{f}_{\boldsymbol{x}} = f_{n_0,\sigma_i^2},\ \text{if } \boldsymbol{x} \in D \cap \otimes_{j=1}^d [x_i^j, x_i^j + \Delta \boldsymbol{x}). \tag{H.3}$$

Therefore, we have

$$\begin{aligned} \int_D W_2^2(\tilde{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) < &\sum_{i=1}^K \int_{D \cap \otimes [x_i^j, x_i^j + \Delta x]} 2\big(W_2^2(\tilde{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}_i}) \\ &+ W_2^2(f_{\boldsymbol{x}}, f_{\boldsymbol{x}_i})\big) \gamma(\mathrm{d}\boldsymbol{x}) = \epsilon. \end{aligned} \tag{H.4}$$

We denote

$$Y(\epsilon, X) := \sup_{i=1,\ldots,K, s=1,\ldots,n_0} \Big(\|\boldsymbol{b}_{i,s}\|^2 + \|A_{i,s}^T A_{i,s}\|_F^2\Big), \tag{H.5}$$

where $\boldsymbol{b}_{i,s}$ and $A_{i,s}^T A_{i,s}$ are the mean vectors and covariance matrices in Eq. (H.2), respectively. Similar to the proof of Corollary 3.1 in Appendix F, for any $\epsilon_1 > 0$, there exists

$$D(\epsilon_1) := D \cap \big(\cup_{i=1}^K \otimes_{j=1}^d [x_i^j, x_i^j + \Delta x - \epsilon_1]\big) \tag{H.6}$$

such that $\gamma(D - D(\epsilon_1)) := \int_{D - D(\epsilon_1)} 1 \gamma(\mathrm{d}\boldsymbol{x}) \le \epsilon$. Additionally, similar to the estimates Eqs. (F.13) and (F.14), we can find an SNN whose output obeys a distribution $\hat{f}_{\boldsymbol{x}}$ satisfying:

$$\begin{aligned} &W_2^2(\hat{f}_{\boldsymbol{x}}, \tilde{f}_{\boldsymbol{x}_i}) \le 6\epsilon Y(\epsilon, X),\ \boldsymbol{x} \in D^i(\epsilon_1) := \{\boldsymbol{x} \in D(\epsilon_1) | \|\boldsymbol{x}_i - \boldsymbol{x}\| \le \|\boldsymbol{x}_j - \boldsymbol{x}\|\} \\ &W_2^2(\hat{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}_i}) \le 2 \max_i \mathbb{E}_{\boldsymbol{y}_{\boldsymbol{x}} \sim f_{n_0,\sigma_i^2}} [\|\boldsymbol{y}_{\boldsymbol{x}}\|^2] + 2\mathbb{E}_{\boldsymbol{y}_{\boldsymbol{x}} \sim \hat{f}_{\boldsymbol{x}}} [\|\boldsymbol{y}_{\boldsymbol{x}}\|^2] \\ &\qquad \le 10 \max_i \mathbb{E}_{\boldsymbol{y}_{\boldsymbol{x}} \sim f_{n_0,\sigma_i^2}} [\|\boldsymbol{y}_{\boldsymbol{x}}\|^2] \le 10 Y(\epsilon, X),\ x \notin D(\epsilon_1). \end{aligned} \tag{H.7}$$

Therefore, we have

$$\begin{aligned} \int_D W_2^2(\hat{f}_{\boldsymbol{x}}, \tilde{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) &\le \int_{D(\epsilon_1)} W_2^2(\hat{f}_{\boldsymbol{x}}, \tilde{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) + \int_{D - D(\epsilon_1)} W_2^2(\hat{f}_{\boldsymbol{x}}, \tilde{f}_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) \\ &= 6\epsilon Y(\epsilon, X) + 10\epsilon Y(\epsilon, X) = 16\epsilon Y(\epsilon, X), \end{aligned} \tag{H.8}$$

and

$$\begin{aligned} \int_D W_2^2(\hat{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}}) \gamma(\mathrm{d}\boldsymbol{x}) &\le 2 \int_D \big(W_2^2(\hat{f}_{\boldsymbol{x}}, \tilde{f}_{\boldsymbol{x}}) + W_2^2(\tilde{f}_{\boldsymbol{x}}, f_{\boldsymbol{x}})\big) \gamma(\mathrm{d}\boldsymbol{x}) \\ &\le 32 Y(\epsilon, X)\epsilon + 2\epsilon. \end{aligned} \tag{H.9}$$

In Eq. (H.9), $Y(\epsilon, X)$ also depends on ϵ and the choice of the grid point set $X = \{\boldsymbol{x}_i\}_{i=1}^K$. Specifically, as $\epsilon \to 0$, if we can design a grid point set $\{\boldsymbol{x}\}_{i=1}^{K(\epsilon)}$ such that the quantity $\epsilon \cdot Y(\epsilon, X) \to 0$ (*e.g.*, when $\boldsymbol{y}$ is uniformly bounded for any $\boldsymbol{x} \in D$, by the construction of f_{n_0,σ_i^2} in Appendix G, $\sup_{i=1,\ldots,K, s=1,\ldots,n_0} \Big(\|\boldsymbol{b}_{i,s}\|^2 + \|A_{i,s}^T A_{i,s}\|_F^2\Big)$ in Eq. H.5 is uniformly bounded and its upper bound is independent of the choice of the grid point set), then there exists an SNN model in Fig. 1 such that the distribution of its output can approximate $f_{\boldsymbol{x}}$, the probability distribution of $\boldsymbol{y}$ given the input $\boldsymbol{x}$, in the squared W_2 sense.

Appendix I. Default training settings

We list the hyperparameters and settings for training the SNN model in Fig. 1 of each example in Table I.3 below.

Table I.3
Training settings for each example.

Hyperparameters	Example 4.1	Example 4.2	Example 4.3	Example 4.4
Gradient descent method	Adam	Adam	Adam	Adam
Learning rate	0.001	0.0005	0.0003	0.001
Weight decay	0.01	0	0.005	0.02
No. of epochs	500	400	2000	400
No. of training trajectories	200	200	400	300
Hidden layers	1	3	3	2
Activation function	ELU	ELU	ELU	ReLU
Neurons in each layer	40	50	10	50
time step Δt	0.1	0.1	0.05	0.1
Number of timesteps N_T	81	21	41	21
Initialization for biases	2	$\mathcal{N}(0, 0.01)$ ($\mathcal{N}((1.5, 0.5)^T, 0.1 I_2)$ for the output layer)	$\mathcal{N}(0, 0.03^2)$	0.01
Initialization for means of weights	$\mathcal{N}(0, 0.01^2)$	$\mathcal{N}(0, 0.01^2)$	$\mathcal{N}(0, 0.03^2)$	$\mathcal{N}(0, 0.01^2)$
Initialization for variances of weights	$\mathcal{N}(0, 0.01^2)$	$\mathcal{N}(0, 0.01^2)$	$\mathcal{N}(0, 0.03^2)$	$\mathcal{N}(0, 0.01^2)$

Appendix J. Definitions of different loss metrics

Below, we provide descriptions and definitions for the different loss functions used for comparison in this study. In the following, N denotes the number of samples, and $t_i = i\frac{T}{N_T}, i = 0, \ldots, N_T$ denotes a uniform mesh in $[0, T]$.

1. A scaled local time-decoupled squared W_2 distance:

$$\frac{1}{N_T} \sum_{i=1}^{N_T} W_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i)) = \frac{1}{N_T N} \sum_{i=1}^{N_T} \sum_{j=1}^{N} W_2^2\big(\nu_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}(t_i), \hat{\nu}_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}(t_i)\big), \tag{J.1}$$

where $W_{2,\delta}^{2,\mathrm{e}}(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))$ is the local squared W_2 distance defined in Eq. (2.4). $\nu_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}(t)$ and $\hat{\nu}_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}(t)$ are the empirical conditional probability distributions of $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}$ at time t conditioned on $|\boldsymbol{X}(0) - \boldsymbol{X}_j(0)| \le \delta$ and $|\hat{\boldsymbol{X}}(0) - \boldsymbol{X}_j(0)| \le \delta$, respectively ($\boldsymbol{X}_j(0)$ denotes the initial state of the jth trajectory in the training set). $W_2^2(\nu_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}, \hat{\nu}_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}})$ is estimated by

$$W_2^2\Big(\nu_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}, \hat{\nu}_{\boldsymbol{X}_{0,j},\delta}^{\mathrm{e}}\Big) \approx \texttt{ot.emd2}\Big(\frac{1}{N_j}\vec{\boldsymbol{I}}_{N_j}, \frac{1}{N_j}\vec{\boldsymbol{I}}_{N_j}, \boldsymbol{C}_j(t_i)\Big), \tag{J.2}$$

where `ot.emd2` is the function for solving the earth movers distance problem in the `POT` package of Python in Flamary et al. (2021). N_j is

the number of elements in the set $X_j := \left\{ \{\boldsymbol{X}_i\}_{t=0}^{T} \middle| \|\boldsymbol{X}_i(0) - \boldsymbol{X}_j(0)\| \le \delta, i = 1, \ldots, N \right\}$, $\vec{\boldsymbol{I}}_{N_j}$ is an N_j-dimensional vector whose elements are all 1, and $\boldsymbol{C}_j(t_i) \in \mathbb{R}^{N_j \times N_j}$ is a matrix with entries $(\boldsymbol{C}_j(t_i))_{sr} = \|\boldsymbol{X}_s(t_i) - \hat{\boldsymbol{X}}_r(t_i)\|^2$. $\boldsymbol{X}_s(t_i)$ and $\hat{\boldsymbol{X}}_r(t_i)$ are the states of the sth ground truth trajectory at time t_i in the set X_j and the states of the rth predicted trajectory at time t_i in the set $\hat{X}_j := \left\{ \{\hat{\boldsymbol{X}}_i\}_{t=0}^{T} \middle| \|\hat{\boldsymbol{X}}_i(0) - \boldsymbol{X}_j(0)\| \le \delta, i = 1, \ldots, N \right\}$, respectively.

2. A scaled time-decoupled squared W_2 distance

$$\tilde{W}_2^2(\boldsymbol{X}, \hat{\boldsymbol{X}}) \approx \frac{1}{N_T} \sum_{i=1}^{N_T} W_2^2(\nu^{\mathrm{e}}(t_i), \hat{\nu}^{\mathrm{e}}(t_i)),$$

where $\nu^{\mathrm{e}}(t_i)$ and $\hat{\nu}^{\mathrm{e}}(t_i)$ are the empirical distributions of $\boldsymbol{X}(t_i)$ and $\hat{\boldsymbol{X}}(t_i)$, respectively. It is estimated by

$$W_2^2(\nu_N^{\mathrm{e}}(t_i), \hat{\nu}_N^{\mathrm{e}}(t_i)) \approx \texttt{ot.emd2}\Big(\frac{1}{N}\vec{\boldsymbol{I}}_N, \frac{1}{N}\vec{\boldsymbol{I}}_N, \boldsymbol{C}(t_i)\Big), \tag{J.3}$$

where `ot.emd2` is the function for solving the earth movers distance problem in the `ot` package of Python in Flamary et al. (2021). N is the number of ground truth and predicted samples, $\vec{\boldsymbol{I}}_N$ is an N-dimensional vector whose elements are all 1, and $\boldsymbol{C}(t_i) \in \mathbb{R}^{N \times N}$ is a matrix with entries $(\boldsymbol{C}(t_i))_{sj} = \|\boldsymbol{X}_s(t_i) - \hat{\boldsymbol{X}}_j(t_i)\|^2$. $\boldsymbol{X}_s(t_i)$ and $\hat{\boldsymbol{X}}_j(t_i)$ are the states of the sth ground truth trajectory at time t_i and the states of the jth predicted trajectory at time t_i, respectively.

3. MMD (maximum mean discrepancy) (Li et al., 2015):

$$\begin{aligned}\mathrm{MMD}(\{\boldsymbol{X}\}, \{\hat{\boldsymbol{X}}\}) = &\frac{1}{N_T}\sum_{i=1}^{N_T} \mathbb{E}[K(\{\boldsymbol{X}(t_i)\}, \{\boldsymbol{X}(t_i)\})] \\ &- 2\mathbb{E}[K(\{\boldsymbol{X}(t_i)\}, \{\hat{\boldsymbol{X}}(t_i)\})] + \mathbb{E}[K(\{\hat{\boldsymbol{X}}(t_i)\}, \{\hat{\boldsymbol{X}}(t_i)\})],\end{aligned} \tag{J.4}$$

where K is the standard radial basis function (or Gaussian kernel) with the multiplier equal to 2 and the number of kernels equal to 5. $\{\boldsymbol{X}(t_i)\}$ and $\{\hat{\boldsymbol{X}}(t_i)\}$ denote the sets of ground truth observations and predicted trajectories at time t_i, respectively.

4. Mean squared error (MSE):

$$\mathrm{MSE}(\boldsymbol{X}, \hat{\boldsymbol{X}}) = \frac{1}{N_T N}\sum_{i=1}^{N_T}\sum_{s=1}^{N} \|\boldsymbol{X}_s(t_i) - \hat{\boldsymbol{X}}_s(t_i)\|^2. \tag{J.5}$$

$\boldsymbol{X}_s(t_i)$ and $\hat{\boldsymbol{X}}_s(t_i)$ are the states of the sth ground truth trajectory at time t_i and the states of the sth predicted trajectory at time t_i, respectively.

5. Mean^2 + Var loss function:

$$\begin{aligned}(\mathrm{Mean}^2 + \mathrm{Var})(\boldsymbol{X}, \hat{\boldsymbol{X}}) = &\frac{1}{N_T}\sum_{i=1}^{N_T}\Bigg(\frac{1}{N}\sum_{s=1}^{N}\|\boldsymbol{X}_s(t_i) - \hat{\boldsymbol{X}}_s(t_i)\|^2 \\ &+ |\mathrm{Var}(\boldsymbol{X}(t_i)) - \mathrm{Var}(\hat{\boldsymbol{X}}(t_i))|\Bigg)\end{aligned} \tag{J.6}$$

where

$$\mathrm{Var}(\boldsymbol{X}(t_i)) = \sum_{s=1}^{N}\Big\|\boldsymbol{X}_s(t_i) - \sum_{i=1}^{N}\frac{1}{N}\boldsymbol{X}_i(t_i)\Big\|^2. \tag{J.7}$$

$\boldsymbol{X}_s(t_i)$ and $\hat{\boldsymbol{X}}_s(t_i)$ are the states of the sth ground truth trajectory at time t_i and the states of the sth predicted trajectory at time t_i, respectively.

Appendix K. Using a neural network to approximate the diffusion function in Example 4.4

Here, we apply a feedforward neural network with deterministic weights and biases to approximate the diffusion function in a jump-diffusion process when the form of the diffusion function is unknown. We consider the following jump-diffusion process:

$$\begin{aligned}&\mathrm{d}X_t = 0.05\mathrm{d}t + \sigma_0\sqrt{|X_t|}\mathrm{d}B_t + \int_U \xi X_t \mathrm{d}\tilde{N}(\nu(\mathrm{d}\xi)\mathrm{d}t), \quad t \in [0, 2], \\ &\xi \sim \mathcal{N}(\beta_0, \sigma_1^2), \quad X_0 \sim \mathcal{N}(2, \sigma_2^2).\end{aligned} \tag{K.1}$$

In Eq. (K.1), σ_0 is a positive constant, and B_t and $\tilde{N}_t$ are an independent Wiener process and a compensated Poisson process, respectively. We use the following approximate jump-diffusion process to approximate Eq. (K.1):

$$\mathrm{d}\hat{X}_t = 0.05\mathrm{d}t + \hat{\sigma}(\hat{X}_t)\mathrm{d}\hat{B}_t + \int_U \hat{\xi}\hat{X}_t \mathrm{d}\hat{N}(\nu(\mathrm{d}\xi)\mathrm{d}t), \quad t \in [0, 2], \quad \hat{X}_0 = X_0. \tag{K.2}$$

In Eq. (K.2), $\hat{\sigma}(\hat{X}_t)$ is a deterministic feedforward parameterized neural network that takes the state $\hat{X}_t$ as the input. We aim to approximate the ground truth diffusion function $|\sigma(\boldsymbol{X}_t)| := |\sigma_0|\sqrt{|X_t|}$ in Eq. (K.1) using the approximate $|\hat{\sigma}(X_t)|$ in Eq. (K.2). $\hat{\xi}$ is the output of an SNN when the input is 1, which aims at approximating the distribution of ξ, and $\hat{\boldsymbol{B}}_t$ is another Wiener process independent of the Wiener process B_t in Eq. (K.1) while $\hat{N}$ is another compensated Poisson process independent of $B_t, \hat{B}_t$ and $\tilde{N}_t$. We train both the deterministic neural network $\hat{\sigma}(\hat{X}_t)$ and the SNN that approximates the distribution of ξ simultaneously by minimizing our local time-decoupled squared W_2 loss function Eq. (2.6). The parameterized neural network $\hat{\sigma}(\hat{X}_t)$ consists of two hidden layers with fifty neurons in each layer. All other hyperparameters are the same as those used in Example 4.4, described in Table I.3. We vary the variance of ξ as well as the value of σ_0 when reconstructing the diffusion function and the distribution of ξ in the jump function.

The trajectories generated by the reconstructed jump-diffusion process align well with ground truth trajectories (shown in Fig. K.7(a)). Furthermore, the error in the reconstructed diffusion function, as well as the error in the reconstructed distribution of $\hat{\xi}$, is small. Thus, when

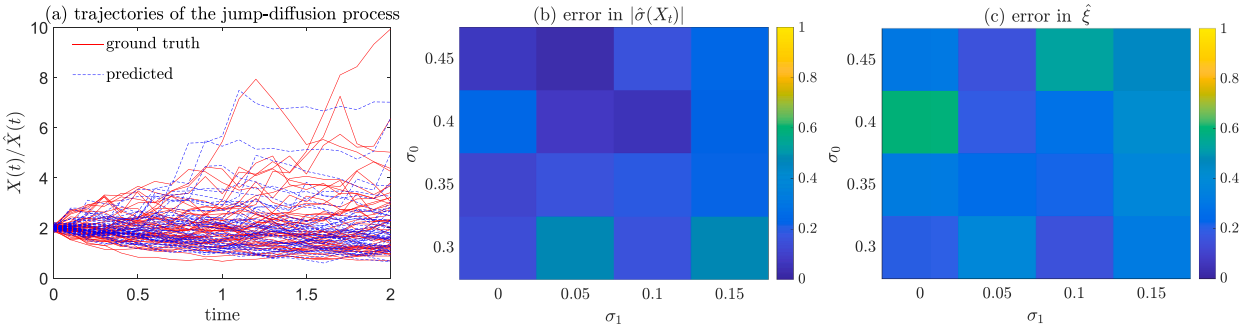

Fig. K.7. (a) Ground truth trajectories versus the reconstructed trajectories when $\sigma_0 = 0.3, \beta_0 = 0.3, \sigma_1 = 0.15, \sigma_2 = 0.1$, and $\delta = 0.1$ in the loss function Eq. (2.6). For clarity, 50 ground truth trajectories and 50 reconstructed trajectories are plotted. (b) The error in the reconstructed diffusion function $\hat{\sigma}$. Here, the error denotes the relative squared L^2 error: $\frac{\int_0^T (|\hat{\sigma}(X_t)| - \sigma_0\sqrt{|X_t|})^2 \mathrm{d}t}{\int_0^T (\sigma_0\sqrt{|X_t|})^2 \mathrm{d}t}$. (c) The error in the reconstructed distribution of $\hat{\xi}$ (Eq. (4.1)). In (b) and (c), the values of other parameters are: $\beta_0 = 0.3, \sigma_2 = 0.1$, and the size of neighborhood $\delta = 0.1$ in the loss function Eq. (J.1). Errors are the average error over three repeated experiments.

the form of the diffusion function is unknown while the diffusion function itself is deterministic, we could consider using a deterministic feedforward neural network to reconstruct it while using an SNN (Fig. 1) to simultaneously reconstruct the distribution of ξ in the jump magnitude function in Eq. (K.1) by minimizing the loss function Eq. (J.1).

References

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214–223). PMLR.

Balasubramanian, K., Goldstein, L., Ross, N., & Salim, A. (2024). Gaussian random field approximation via Stein's method with applications to wide random neural networks. *Applied and Computational Harmonic Analysis*, *72*, 101668.

Bernton, E., Jacob, P. E., Gerber, M., & Robert, C. P. (2019). On parameter estimation with the Wasserstein distance. *Information and Inference: A Journal of the IMA*, *8*(4), 657–676.

Blanchet, J., & Kang, Y. (2021). Sample out-of-sample inference based on Wasserstein distance. *Operations Research*, *69*(3), 985–1013.

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. In *International conference on machine learning* (pp. 1613–1622). PMLR.

Bayesian neural network derivation (2020). https://www.zhihu.com/tardis/bd/art/263053978.

Böhm, V., Lanusse, F., & Seljak, U. (2019). Uncertainty quantification with generative models. arXiv preprint arXiv:1910.10046

Boukraichi, H., Akkari, N., Casenave, F., & Ryckelynck, D. (2022). Uncertainty quantification in a mechanical submodel driven by a Wasserstein-GAN. *IFAC-PapersOnLine*, *55*(20), 469–474.

Brunton, S. L., Proctor, J. L., & Kutz, J. N. (2016). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, *113*(15), 3932–3937. https://doi.org/10.1073/pnas.1517384113

Carrassi, A., Bocquet, M., Bertino, L., & Evensen, G. (2018). Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *Wiley Interdisciplinary Reviews: Climate Change*, *9*(5), e535. https://doi.org/10.1002/wcc.535

Caruso, A., Füth, M., Alvarez-Sánchez, R., Belli, S., Diack, C., Maass, K. F., Schwab, D., Kettenberger, H., & Mazer, N. A. (2019). Ocular half-life of intravitreal biologics in humans and other species: Meta-analysis and model-based prediction. *Molecular Pharmaceutics*, *17*(2), 695–709.

Chappelow, A. V., & Kaiser, P. K. (2008). Neovascular age-related macular degeneration: potential therapies. *Drugs*, *68*, 1029–1036.

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, *31*.

Chib, S., & Greenberg, E. (2002). Bayesian inference for regression models with ARIMA errors. *Journal of Econometrics*, *109*, 385–408. https://doi.org/10.1016/S0304-4076(01)00136-7

Clement, P., & Desch, W. (2008). An elementary proof of the triangle inequality for the Wasserstein metric. *Proceedings of the American Mathematical Society*, *136*(1), 333–339.

Crawshaw, J., Gaffney, E., Gertz, M., Maini, P., & Caruso, A. (2025). Modelling the ocular pharmacokinetics and pharmacodynamics of ranibizumab for improved understanding and data collection strategies in ocular diseases. *Investigative Ophthalmology & Visual Science*, *66*(6), 20.

Cresswell, W. (2015). Maximum likelihood estimation for dynamical models: A guide to implementation. *Journal of Theoretical Biology*, *382*, 123–132. https://doi.org/10.1016/j.jtbi.2015.06.028

Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, *26*.

Deng, Y., Shao, S. S., Mogilner, A., & Xia, M. (2025). Adaptive hyperbolic-cross-space mapped Jacobi method on unbounded domains with applications to solving multidimensional spatiotemporal integrodifferential equations. *Journal of Computational Physics*, *520*, 113492.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T. H., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., …Vayer, T. (2021). POT: Python optimal transport. *Journal of Machine Learning Research*, *22*(78), 1–8.

Fournier, N., & Guillin, A. (2015). On the rate of convergence in Wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, *162*(3), 707–738.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, *48*, 1050–1059. https://proceedings.mlr.press/v48/gal16.html.

Gonon, L., Grigoryeva, L., & Ortega, J.-P. (2023). Approximation bounds for random neural networks and reservoir systems. *The Annals of Applied Probability*, *33*(1), 28–69.

Grecksch, W., & Kloeden, P. E. (1996). Time-discretised Galerkin approximations of parabolic stochastic PDEs. *Bulletin of the Australian Mathematical Society*, *54*(1), 79–85.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of Wasserstein GANs. *Advances in Neural Information Processing Systems*, *30*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hoffer, E., Hubara, I., & Soudry, D. (2017). Train longer, generalize better: closing the generalization gap in large batch training of neural networks. *Advances in Neural Information Processing Systems*, *30*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Hutton-Smith, L. A., Gaffney, E., Byrne, H. M., Caruso, A., Maini, P., & Mazer, N. A. (2018). Theoretical insights into the retinal dynamics of VEGF in patients treated with ranibizumab, based on an ocular pharmacokinetic/pharmacodynamic model. *Molecular Pharmaceutics*, *15*(7), 2770–2784.

Hutton-Smith, L. A., Gaffney, E. A., Byrne, H. M., Maini, P. K., Schwab, D., & Mazer, N. A. (2016). A mechanistic model of the intravitreal pharmacokinetics of large molecules and the pharmacodynamic suppression of ocular vascular endothelial growth factor levels by ranibizumab in patients with neovascular age-related macular degeneration. *Molecular Pharmaceutics*, *13*(9), 2941–2950.

Kloeden, P. E., & Platen, E. (1992). Numerical Solution of Stochastic Differential Equations. Berlin: Springer Nature.

Lamirande, P., Gaffney, E. A., Gertz, M., Maini, P. K., Crawshaw, J. R., & Caruso, A. (2024). A first-passage model of intravitreal drug delivery and residence time–influence of ocular geometry, individual variability, and injection location. *Investigative Ophthalmology & Visual Science*, *65*(12), 21.

Leshno, M., Lin, V. Y., Pinkus, A., & Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, *6*(6), 861–867.

Li, Y., Swersky, K., & Zemel, R. (2015). Generative moment matching networks. In *International conference on machine learning* (pp. 1718–1727). PMLR.

Liu, W., & Röckner, M. (2015). Stochastic Partial Differential Equations: an Introduction. Springer.

Loskot, P., Atitey, K., & Mihaylova, L. (2019). Comprehensive review of models and methods for inferences in bio-chemical reaction networks. *Frontiers in Genetics*, *10*, 549.

Lu, Y., & Lu, J. (2020). A universal approximation theorem of deep neural networks for expressing probability distributions. *Advances in Neural Information Processing Systems*, *33*, 3094–3105.

Manita, O. A., Peletier, M. A., Portegies, J. W., Sanders, J., & Senen-Cerda, A. (2022). Universal approximation in dropout neural networks. *Journal of Machine Learning Research*, *23*(19), 1–46.

Marin, J.-M., Pudlo, P., Robert, C. P., & Ryder, R. J. (2021). Approximate Bayesian computational methods. *Statistics and Computing*, *31*, 239–259. https://doi.org/10.1007/s11222-020-09968-y

Mehri, S., & Scheutzow, M. (2019). A stochastic Gronwall lemma and well-posedness of path-dependent SDEs driven by martingale noise. arXiv preprint arXiv:1908.10646

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, *3*(1–2), 125–144.

Mitchell, P., Liew, G., Gopinath, B., & Wong, T. Y. (2018). Age-related macular degeneration. *The Lancet*, *392*(10153), 1147–1159.

Mullachery, V., Khera, A., & Husain, A. (2018). Bayesian neural networks. arXiv preprint arXiv:1801.07710

Nardini, J. T., & Bortz, D. M. (2019). The influence of numerical error on parameter estimation and uncertainty quantification for advective PDE models. *Inverse Problems*, *35*(6), 065003.

Nardini, J. T., Lagergren, J. H., Hawkins-Daarud, A., Curtin, L., Morris, B., Rutter, E. M., Swanson, K. R., & Flores, K. B. (2020). Learning equations from biological data with limited time samples. *Bulletin of Mathematical Biology*, *82*, 1–33.

Neal, R. M. (2012). Bayesian Learning for Neural Networks (118). Springer Science & Business Media.

NYU HPC (2025). NYU HPC hardware specs. https://sites.google.com/nyu.edu/nyu-hpc/hpc-systems/greene/hardware-specs.

Panaretos, V. M., & Zemel, Y. (2019). Statistical aspects of Wasserstein distances. *Annual Review of Statistics and its Application*, *6*(1), 405–431.

Pathak, J., Lu, Z., Hunt, B., Girvan, M., & Ott, E. (2018). Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Physical Review Letters*, *120*(2), 024102. https://doi.org/10.1103/PhysRevLett.120.024102

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., & Simoncelli, E. P. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, *454*(7207), 995–999.

Platen, E., & Bruti-Liberati, N. (2010). Numerical solution of stochastic differential equations with jumps in finance (vol. 64). Springer Science & Business Media.

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2017). Machine learning of linear differential equations using gaussian processes. *Journal of Computational Physics*, *348*, 683–693. https://doi.org/10.1016/j.jcp.2017.07.050

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, *378*, 686–707.

Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. *Proceedings of the 31st International Conference on Machine Learning (ICML)*, *32*, 1278–1286. https://proceedings.mlr.press/v32/rezende14.html.

Roda, W. C., Varughese, M. B., Han, D., & Li, M. Y. (2020). Why is it difficult to accurately predict the COVID-19 epidemic? *Infectious Disease Modelling*, *5*, 271–281.

Senan, S., Ali, M. S., Vadivel, R., & Arik, S. (2017). Decentralized event-triggered synchronization of uncertain Markovian jumping neutral-type neural networks with mixed delays. *Neural Networks*, *86*, 32–41.

Shen, J., Tang, T., & Wang, L.-L. (2011). Spectral Methods: Algorithms, Analysis and Applications (41). Springer Science & Business Media.

Shen, J., & Wang, L. (2010). Sparse spectral approximations of high-dimensional problems based on hyperbolic cross. *SIAM Journal on Numerical Analysis*, *48*, 1087–1109.

Tang, C., & Salakhutdinov, R. R. (2013). Learning stochastic feedforward neural networks. *Advances in Neural Information Processing Systems*, *26*.

Vadivel, R., Ali, M. S., Alzahrani, F., Cao, J., & Joo, Y. H. (2020). Synchronization of decentralized event-triggered uncertain switched neural networks with two additive time-varying delays. *Nonlinear Analysis: Modelling and Control*, *25*(2), 183–205.

Villani, C. (2009). Optimal Transport: Old and New (338). Springer.

Xia, M., Li, X., Shen, Q., & Chou, T. (2024a). An efficient Wasserstein-distance approach for reconstructing jump-diffusion processes using parameterized neural networks. *Machine Learning: Science and Technology*, *5*, 045052.

Xia, M., Li, X., Shen, Q., & Chou, T. (2024b). Squared Wasserstein-2 distance for efficient reconstruction of stochastic differential equations. arXiv preprint arXiv:2401.11354

Xia, M., & Shen, Q. (2024). A local squared Wasserstein-2 method for efficient reconstruction of models with uncertainty. arXiv preprint arXiv:2406.06825

Yu, T., Yang, Y., Li, D., Hospedales, T., & Xiang, T. (2021). Simple and effective stochastic neural networks. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 3252–3260). (*35*).