IOP Publishing Mach. Learn.: Sci. Technol. 5 (2024) 045052 https://doi.org/10.1088/2632-2153/ad9379

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
28 May 2024

REVISED
3 November 2024

ACCEPTED FOR PUBLICATION
15 November 2024

PUBLISHED
12 December 2024

PAPER

An efficient Wasserstein-distance approach for reconstructing jump-diffusion processes using parameterized neural networks

Mingtao Xia[1,*], **Xiangting Li**[2], **Qijing Shen**[3] and **Tom Chou**[4]

[1] Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, United States of America
[2] Department of Computational Medicine, UCLA, Los Angeles, CA 90095, United States of America
[3] Nuffield Department of Medicine, University of Oxford, Oxford OX2 6HW, United Kingdom
[4] Department of Mathematics, UCLA, Los Angeles, CA 90095, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: xiamingtao@nyu.edu, xiangting.li@ucla.edu, qijing.shen@ndm.ox.ac.uk and tomchou@ucla.edu

Keywords: jump-diffusion process, inverse problem, Wasserstein distance, neural network

Abstract

We analyze the Wasserstein distance (W-distance) between two probability distributions associated with two multidimensional jump-diffusion processes. Specifically, we analyze a temporally decoupled squared W_2-distance, which provides both upper and lower bounds associated with the discrepancies in the drift, diffusion, and jump amplitude functions between the two jump-diffusion processes. Then, we propose a temporally decoupled squared W_2-distance method for efficiently reconstructing unknown jump-diffusion processes from data using parameterized neural networks. We further show its performance can be enhanced by utilizing prior information on the drift function of the jump-diffusion process. The effectiveness of our proposed reconstruction method is demonstrated across several examples and applications.

1. Introduction

Jump-diffusion processes are widely used across many disciplines such as finance [1–3], biology [4], epidemiology [5], and so on. A d-dimensional jump-diffusion process may be written in the following form [6]:

$$\mathrm{d}\boldsymbol{X}(t) = \boldsymbol{f}(\boldsymbol{X}(t),t)\,\mathrm{d}t + \boldsymbol{\sigma}(\boldsymbol{X}(t),t)\,\mathrm{d}\boldsymbol{B}_t + \int_U \boldsymbol{\beta}(\boldsymbol{X}(t),\xi,t)\,\tilde{N}(\mathrm{d}t,\nu(\mathrm{d}\xi)). \tag{1}$$

Here, $\boldsymbol{X}(t)\in\mathbb{R}^d$ is a d-dimensional jump-diffusion process and $\boldsymbol{B}_t := (B_{1,t},\ldots,B_{m,t})$ is a standard m-dimensional white noise; $\tilde{N}$ is a compensated Poisson process of intensity $\nu(\mathrm{d}\xi)\mathrm{d}t$ independent of $\boldsymbol{B}_t$:

$$\tilde{N}(\mathrm{d}t,\nu(\mathrm{d}\xi)) := N(\mathrm{d}t,\nu(\mathrm{d}\xi)) - \nu(\mathrm{d}\xi)\,\mathrm{d}t, \tag{2}$$

where $N(\mathrm{d}t,\nu(\mathrm{d}\xi))$ is a Poisson process with intensity $\nu(\mathrm{d}\xi)\mathrm{d}t$ and $\nu(\mathrm{d}\xi)$ is a measure defined on $U\subseteq\mathbb{R}$, the measure space of the Poisson process. $N(A,B)$ and $N(C,D)$ are independent if $(A\times B)\cap(C\times D)=\emptyset, A,C\subseteq\mathcal{B}(U)$ and $B,D\subseteq\mathcal{B}([0,T])$. $\mathcal{B}(U)$ and $\mathcal{B}([0,T])$ denote the Borel σ-algebra associated with U and $[0,T]$, respectively. The drift, diffusion, and jump functions are defined by

$$\begin{aligned}\boldsymbol{f} &:= (f_1(\boldsymbol{X},t),\ldots,f_d(\boldsymbol{X},t)) \in C\left(\mathbb{R}^d\times[0,T],\mathbb{R}^d\right),\\ \boldsymbol{\sigma} &:= \left(\sigma_{i,j}(\boldsymbol{X},t)\right) \in C\left(\mathbb{R}^d\times[0,T],\mathbb{R}^{d\times m}\right),\\ \boldsymbol{\beta} &:= (\beta_i(\boldsymbol{X},\xi,t)) \in C\left(\mathbb{R}^d\times U\times[0,T],\mathbb{R}^d\right),\end{aligned} \tag{3}$$

respectively. Specifically, if $U=\{1,\ldots,n\}$, then equation (1) becomes

$$\mathrm{d}X_i(t) = f_i(\boldsymbol{X}(t),t)\,\mathrm{d}t + \sum_{j=1}^{m}\sigma_{i,j}(\boldsymbol{X}(t),t)\,\mathrm{d}B_{j,t} + \sum_{s=1}^{n}\beta_{i,s}(\boldsymbol{X}(t),s,t)\,\tilde{N}_s(\mathrm{d}t,\nu(s)) \tag{4}$$

for $i = 1, .., d$. Here, each $\tilde{N}_s$ is a compensated Poisson process with intensity $\nu(s)\mathrm{d}t$. $\tilde{N}_{s_1}$ and $\tilde{N}_{s_2}$ are independent if $s_1 \neq s_2$. When $\boldsymbol{\beta} \equiv 0$, equation (1) reduces to the pure diffusion process.

In this paper, we study the problem of reconstructing a jump-diffusion process equation (1) or equation (4) from observed data $\boldsymbol{X}(t)$ at different time points by using a different jump-diffusion process

$$\mathrm{d}\hat{\boldsymbol{X}}(t) = \hat{\boldsymbol{f}}(\boldsymbol{X}(t), t)\,\mathrm{d}t + \hat{\boldsymbol{\sigma}}\left(\hat{\boldsymbol{X}}(t), t\right)\mathrm{d}\hat{\boldsymbol{B}}_t + \int_U \hat{\boldsymbol{\beta}}\left(\hat{\boldsymbol{X}}(t), \xi, t\right)\hat{N}(\mathrm{d}t, \nu(\mathrm{d}\xi)) \tag{5}$$

to approximate equation (1). In equation (5), $\hat{\boldsymbol{B}}_t$ is a m-dimensional standard Brownian motion that is independent of $\boldsymbol{B}_t$ and $\tilde{N}$ in equation (1); $\hat{N}(\mathrm{d}t, \nu(\mathrm{d}\xi))$ is a compensated Poisson process of intensity $\mathrm{d}\nu(\xi)\mathrm{d}t$ and independent of $\boldsymbol{B}_t$, $\tilde{N}$ in equation (1) as well as $\hat{\boldsymbol{B}}_t$. Specifically, we are interested in reconstructing the jump-diffusion process equation (1) with little or no prior information on the drift, diffusion, and jump functions $\boldsymbol{f}, \boldsymbol{\sigma}$, and $\boldsymbol{\beta}$. To reconstruct or approximate equation (1) using equation (5), we wish to find small errors in the drift, diffusion, and jump functions, i.e. to find $\hat{\boldsymbol{f}}, \hat{\boldsymbol{\sigma}}$, and $\hat{\boldsymbol{\beta}}$ such that $\boldsymbol{f} - \hat{\boldsymbol{f}}$, $\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}$ are small.

Thus far, most studies related to jump-diffusion processes have focused on the forward-type problem of efficient simulation of a jump-diffusion process given coefficients [7, 8]. There are also several studies on the statistical properties of jump-diffusion processes such as their first passage times [9, 10]. While there has been some research into the inverse problem of reconstructing a general pure diffusion process, there has been little work on reconstructing unknown jump-diffusion processes from sample trajectories. However, reconstructing jump-diffusion processes is important for understanding stochastic dynamics in complex systems arising in physics, biology, finance, and other disciplines, especially those characterized by discontinuities and intrinsic noise. Examples include the Boltzmann equation for particle interactions [11], material science such as supercooled water [12], quantum dynamics as represented by the Lindblad equation [13], and muscle contraction modeling in cellular biophysics [14]. These applications underscore the critical role of jump-diffusion process reconstruction in advancing knowledge across physical, engineering, and biomedical domains.

So far, two main strategies for reconstruction have been proposed. First, regression methods are applied to determine unknown parameters if the forms of drift, diffusion, and jump functions ($\boldsymbol{f}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$ in equation (1)) are known. Unknown parameters in these functions can then be determined from data [15, 16]. Another strategy for reconstructing a jump-diffusion process is to calculate the empirical probability density function $p(\boldsymbol{X}, t)$ from observation data $\boldsymbol{X}(t)$ and then reconstruct the integrodifferential equation satisfied by $p(\boldsymbol{X}, t)$ [17]. Yet, this method requires a large number of observations at different time points to obtain a good empirical approximation of the density function $p(\boldsymbol{X}, t)$. Recently, a Wasserstein-generative-adversarial-network(WGAN)-based method was proposed for reconstructing the jump-diffusion process equation (1) [4]. However, training a WGAN can be intricate and computationally expensive.

Recent advancements in machine learning make it possible to use parameterized neural networks (NNs) for representing $\hat{\boldsymbol{f}}, \hat{\boldsymbol{\sigma}}$, and $\hat{\boldsymbol{\beta}}$ in equation (5) which approximate $\boldsymbol{f}, \boldsymbol{\sigma}$, and $\boldsymbol{\beta}$ in equation (1). For example, a recent `torchsde` package in Python [18] models pure diffusion processes (SDEs with Brownian noise) by using parameterized neural networks. These methods have been used in the reconstruction of diffusion processes. For example, in [19], a deep Gaussian latent model has been applied for reconstructing a pure diffusion process; [20] uses the neural SDE model to reconstruct a stochastic differential equation with Brownian noise by minimizing a KL-divergence-based loss function. In [21, 22], generative adversarial networks were used to reconstruct general stochastic differential equations including a Brownian motion noise term without requiring prior knowledge of the specific forms of the drift or diffusion functions.

A key challenge in reconstructing jump-diffusion processes is to properly quantify the discrepancies between the distributions generated by the true jump-diffusion process equation (1) and the approximate jump-diffusion process equation (5) using only sample trajectories. Although loglikelihood-based methods such as KL divergence are often used in the probabilistic modeling, they are not suitable for evaluating the distance between empirical distributions over the space of functions. This is because the likelihood of unobserved trajectories in the empirical distribution is 0 and its log-likelihood is undefined. For finite dimensional data, one can use Gaussian distribution to smoothen the empirical distribution and then calculate the smoothened log-likelihood. However, this method is not suitable for infinite dimensional data such as the trajectories of jump-diffusion processes. By contrast, the Wasserstein distance can effectively measure discrepancies between probability measures defined over a metric space and is readily differentiable with respect to the parameters of the neural networks [23, 24]. Consequently, an efficient squared-Wasserstein-distance-based method for reconstructing pure diffusion processes from data, without the need to specify forms for the drift and diffusion, was recently proposed [25]. General jump-diffusion processes are distinct from pure diffusion processes because the trajectories of jump-diffusion processes are discontinuous.

Although some recent work analyzes a smooth Wasserstein distance between two distributions associated with two 1D jump-diffusion processes at a given time [26], it remains unclear whether the Wasserstein distance can also be employed in the reconstruction of an unknown jump-diffusion process from data.

1.1. Contribution

In this paper, we analyze the W-distance between two probability distributions associated with two multidimensional jump-diffusion processes equations (1) and (5). We then show that a temporally decoupled squared Wasserstein distance can serve as effective **upper and lower error bounds** on the errors in the drift, diffusion, and jump functions $\boldsymbol{f}-\hat{\boldsymbol{f}}, \boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$ in equations (1) and (5), respectively. This temporally decoupled squared Wasserstein distance can be effectively evaluated using finite-sample observations at discrete time points. Thus, we propose using this temporally decoupled squared W_2-distance in the reconstruction of general jump-diffusion processes with the help of parameterized neural networks. Our method directly solves the inverse-type problem of reconstructing jump-diffusion processes from time-series data. Furthermore, we explore how prior information on the drift function enhances the performance of our temporally decoupled squared Wasserstein distance method. Our results greatly extend the results in [25] (reconstructing 1D pure diffusion processes) to allow for the reconstruction of multidimensional jump-diffusion processes. Specifically, we

- prove that the W-distance is a lower bound for the errors $\boldsymbol{f}-\hat{\boldsymbol{f}}, \boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$. Thus, minimizing the W-distance is necessary for reconstructing the multidimensional jump-diffusion process equation (1).
- analyze a temporally decoupled squared W-distance defined in [25] and show that it can be efficiently evaluated by finite-sample empirical distributions. Thus, it is suitable as a loss function for reconstructing equation (1) using parameterized neural networks.
- conduct numerical experiments to demonstrate the efficacy of using the temporally decoupled squared Wasserstein distance in the reconstruction of jump-diffusion processes.

The advantages of our proposed temporally decoupled squared W_2-distance-based reconstruction method for jump-diffusion processes include:

1. It can directly reconstruct jump-diffusion processes such as equation (1) from observed temporal trajectories by using parameterized neural networks to approximate the drift, diffusion, and jump functions $\boldsymbol{f}, \boldsymbol{\sigma}$, and $\boldsymbol{\beta}$, respectively. These parameterized neural networks are straightforwardly trained by minimization of a simple temporally decoupled squared W_2-distance loss function, which can be directly evaluated using the `POT` package [27]
2. Our temporally decoupled squared Wasserstein method outperforms several other benchmark methods, such as the minimization of other commonly used loss functions in UQ, e.g. maximum-mean-discrepancy, mean square error, mean^2+var as well as a WGAN method.
3. Based on our empirical results, prior information on the drift function can further increase accuracy in the reconstruction of the diffusion and jump functions in equation (1). Such prior information enables one to accurately reconstruct a jump-diffusion process with only several hundred observed trajectories.

1.2. Organization

In section 2, we analyze how the W-distance between the probability measures associated with solutions to two jump-diffusion processes equations (1) and (5) can be a lower bound of the errors in the reconstructed drift, diffusion, and jump functions $\boldsymbol{f}-\hat{\boldsymbol{f}}, \boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$. In section 3, we analyze a temporally decoupled squared W_2-distance and show how it can be more effectively evaluated than the squared W_2 distance. Specifically, the temporally decoupled squared W_2 distance is smaller than the W_2 distance analyzed in section 2 while providing an upper bound of the errors in the reconstructed drift, diffusion, and jump functions. Thus, sections 2 and 3 together show that our temporally decoupled squared W_2 distance provides both upper and lower error bounds. In section 4, numerical experiments are carried out to compare our proposed jump-diffusion process reconstruction methods with other methods for reconstructing different jump-diffusion processes. Additionally, we show how prior information on the drift function of the ground truth jump-diffusion process equation (1) improves the reconstruction of the diffusion and jump functions in equation (1). In section 5, we summarize our proposed jump-diffusion process reconstruction approach and suggest potential future directions.

2. The W-distance between the probability measures associated with the jump-diffusion processes in equations (1) and (5)

In this section, we shall show how the W-distance between the probability measures associated with the two jump-diffusion processes equations (1) and (5) can serve as a lower bound for the errors $\boldsymbol{f}-\hat{\boldsymbol{f}}$, $\boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$. First, we specify the assumptions on the jump-diffusion processes in equations (1) and (5).

Assumption 2.1. We assume that the jump-diffusion processes defined in equation (1) satisfy the following conditions:

(i) For each non-increasing sequence $A_i \in \mathcal{B}(U)$ converging to the empty set $\emptyset$, $\mathbb{E}\left[|\tilde{N}(t,A)|^2\right] \to 0, \forall t \geqslant 0$.
(ii) $\tilde{N}(t,A)$ is a càdlàg martingale for all $A \in \mathcal{B}(U), t>0$, and $\mathbb{E}[|\tilde{N}(t,U)|^2] < \infty$.
(iii) $\tilde{N}\big(\mathrm{d}t, \nu(\mathrm{d}\xi)\big)$ is an orthogonal martingale measure with intensity $\mathrm{d}t\cdot\nu(\mathrm{d}\xi)$, i.e. for any $A, B \in \mathcal{B}(U)$ and $t_1 \leqslant t_2, t_3 \leqslant t_4$ and any $\beta_1(\xi,t), \beta_2(\xi,t) \in L^2([0,T]\times U)$ (the measure on U is ν), we have

$$\mathbb{E}\left[\int_{t_1}^{t_2}\int_A \beta_1(\xi,t)\tilde{N}(\mathrm{d}t,\nu(\mathrm{d}\xi))\cdot\int_{t_3}^{t_4}\int_B \beta_2(\xi,t)\tilde{N}(\mathrm{d}t,\nu(\mathrm{d}\xi))\right] = \int_{[t_1,t_2]\cap[t_3,t_4]}\int_{A\cap B}\beta_1(\xi,t)\beta_2(\xi,t)\nu(\mathrm{d}\xi)\mathrm{d}t. \tag{6}$$

(iv) Trajectories generated from both jump-diffusion processes, equations (1) and (5), reside in the space $L^2([0,T],\mathbb{R}^d)$.
(v) The drift, diffusion, and jump functions are all uniformly Lipschitz continuous, *i.e.*, there exists three positive constants $\bar{f},\overline{\sigma},\overline{\beta} < \infty$ such that $\forall \boldsymbol{X}^1 = (X_1^1,\ldots,X_d^1), \forall \boldsymbol{X}^2 = (X_1^2,\ldots,X_d^2) \in \mathbb{R}^d$,

$$\begin{aligned}
\left|f_i(\boldsymbol{X}^1,t) - f_i(\boldsymbol{X}^2,t)\right| &\leqslant \frac{\bar{f}}{d}\sum_{i=1}^d \left|X_i^1 - X_i^2\right|,\ \forall i = 1,\ldots,d\\
\left|\sigma_{i,j}(\boldsymbol{X}^1,t) - \sigma_{i,j}(\boldsymbol{X}^2,t)\right| &\leqslant \frac{\overline{\sigma}}{d}\sum_{i=1}^d \left|X_i^1 - X_i^2\right|,\ \forall i = 1,\ldots,d,\ \forall j = 1,\ldots,m,\\
\left|\beta_i(\boldsymbol{X}^1,\xi,t) - \beta_i(\boldsymbol{X}^2,\xi,t)\right| &\leqslant \frac{\overline{\beta}}{d}\sum_{i=1}^d \left|X_i^1 - X_i^2\right|,\ \forall i = 1,\ldots,d.
\end{aligned} \tag{7}$$

Furthermore, we assume that conditions (i)–(iv) also hold for the compensated Poisson process $\hat{N}$ in equation (5), and that condition (v) holds for the drift, diffusion, and jump functions in equation (5).

Now consider the W-distance between the distributions associated with solutions generated from the target jump-diffusion process equation (1) and the approximate jump-diffusion process equation (5), as defined below.

Definition 2.1. For two d-dimensional jump-diffusion processes

$$\boldsymbol{X}(t) = \left(X_1(t),\ldots,X_d(t)\right),\ \hat{\boldsymbol{X}}(t) = \left(\hat{X}_1(t),\ldots,\hat{X}_d(t)\right),\ t\in[0,T], \tag{8}$$

in the separable space $\left(L^2([0,T];\mathbb{R}^d),\|\cdot\|\right)$ with two associated probability distributions $\mu,\hat{\mu}$, respectively, the W_p-distance $W_p(\mu,\hat{\mu})$ for $1\leqslant p\leqslant 2$ is defined as

$$W_p(\mu,\hat{\mu}) := \inf_{\pi(\mu,\hat{\mu})}\mathbb{E}_{(\boldsymbol{X},\hat{\boldsymbol{X}})\sim\pi(\mu,\hat{\mu})}\left[\|\boldsymbol{X}-\hat{\boldsymbol{X}}\|^p\right]^{\frac{1}{p}}. \tag{9}$$

In equation (9), the norm $\|\cdot\|$ is defined as $\|\boldsymbol{X}\| := \left(\int_0^T\sum_{i=1}^d |X_i(t)|^2\mathrm{d}t\right)^{\frac{1}{2}}$ and $\pi(\mu,\hat{\mu})$ iterates over all *coupled* distributions of $\boldsymbol{X}(t),\hat{\boldsymbol{X}}(t)$, defined by the condition

$$\begin{cases}
\boldsymbol{P}_{\pi(\mu,\hat{\mu})}\left(A\times L^2\left([0,T];\mathbb{R}^d\right)\right) = \boldsymbol{P}_{\mu}(A),\\
\boldsymbol{P}_{\pi(\mu,\hat{\mu})}\left(L^2\left([0,T];\mathbb{R}^d\right)\times A\right) = \boldsymbol{P}_{\hat{\mu}}(A),
\end{cases} \qquad \forall A\in\mathcal{B}\left(L^2\left([0,T];\mathbb{R}^d\right)\right), \tag{10}$$

where $\mathcal{B}(L^2([0,T];\mathbb{R}^d))$ denotes the Borel σ-algebra associated with the space of d-dimensional functions in $L^2([0,T];\mathbb{R}^d)$.

To prove that $W_p(\mu,\hat{\mu})$ defined in equation (9) is a lower bound for the errors in the drift, diffusion, and jump functions $\boldsymbol{f}-\hat{\boldsymbol{f}}$, $\boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$, we first prove the following theorem.

Theorem 2.1. *Suppose $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ are two d-dimensional jump-diffusion processes that are determined by equations (1) and (5). We denote*

$$d\tilde{\boldsymbol{X}}(t) = \hat{\boldsymbol{f}}\left(\tilde{\boldsymbol{X}}(t), t\right) + \hat{\boldsymbol{\sigma}}\left(\tilde{\boldsymbol{X}}(t), t\right) d\boldsymbol{B}_t + \int_U \hat{\boldsymbol{\beta}}\left(\tilde{\boldsymbol{X}}(t), \xi, t\right) \tilde{N}(dt, \nu(d\xi)) \tag{11}$$

and assume that

$$\int_0^t \left(X_i\left(s^-\right) - \tilde{X}_i\left(s^-\right)\right)\left(\sigma_{i,j} - \hat{\sigma}_{i,j}\right) dB_{j,t}, \ \int_0^t \int_U \left(X_i\left(s^-\right) - \tilde{X}_i\left(s^-\right)\right)\left(\beta_i - \hat{\beta}_i\right) \tilde{N}(dt, \nu(d\xi)), \tag{12}$$

are martingales for all i, j. Then, the following inequality holds:

$$\mathbb{E}\left[\left|\boldsymbol{X}(t) - \tilde{\boldsymbol{X}}(t)\right|_2^2\right] \leqslant \mathbb{E}\left[H(T)\left|\boldsymbol{X}(0)\right.\right] \exp\left(\left[2\bar{f} + 1 + (2\overline{\sigma} + 1)\, m + (2\overline{\beta} + 1)\nu(U)\right] T\right), \tag{13}$$

where $|\cdot|_2$ denotes the 2-norm of a d-dimensional vector, $\boldsymbol{X}(0)$ is the initial condition, and $H(t)$ is defined as

$$\begin{aligned} H(t) := & \mathbb{E}\left[\sum_{i=1}^{d} \int_0^t \left(f_i\left(\boldsymbol{X}\left(s^-\right), s^-\right) - \hat{f}_i\left(\boldsymbol{X}\left(s^-\right), s^-\right)\right)^2 ds\right] \\ & + \mathbb{E}\left[\sum_{i=1}^{d} \int_0^t \sum_{j=1}^{m} \left(\sigma_{i,j}\left(\boldsymbol{X}\left(s^-\right), s^-\right) - \hat{\sigma}_{i,j}\left(\boldsymbol{X}\left(s^-\right), s^-\right)\right)^2 ds\right] \\ & + \mathbb{E}\left[\sum_{i=1}^{d} \int_0^t \int_U \left(\beta_i\left(\boldsymbol{X}\left(s^-\right), \xi, s^-\right) - \hat{\beta}_i\left(\boldsymbol{X}\left(s^-\right), \xi, s^-\right)\right)^2 \nu(d\xi)\, ds\right]. \end{aligned} \tag{14}$$

The proof to theorem 2.1 is similar to the proof of the stochastic Gronwall lemma (theorem 2.2 in [6]) and is given in appendix A. Theorem 2.1 greatly generalizes the results of theorem 1 in [25], which was developed for analyzing the W_2-distance between two one-dimensional pure diffusion processes. Now, with theorem 2.1, we can analyze the W-distance between two multi-dimensional jump-diffusion processes.

The following corollary establishes the upper bound of the W-distance $W_p(\mu, \hat{\mu}), 1 \leqslant p \leqslant 2$ between μ and $\hat{\mu}$, the two probability distributions associated with jump-diffusion processes equations (1) and (5).

Corollary 2.1. (Upper error bound for the W-distance) The following bound holds for $W_p(\mu, \hat{\mu})$, where μ and $\hat{\mu}$ are the two probability distributions associated with jump-diffusion processes equations (1) and (5)

$$\mathrm{W}_p(\mu, \hat{\mu}) \leqslant \sqrt{\mathrm{T}\mathbb{E}\left[H(T)\left|\boldsymbol{X}(0)\right.\right]} \exp\left(\left[\bar{f} + \tfrac{1}{2} + \left(\overline{\sigma} + \tfrac{1}{2}\right) m + \left(\overline{\beta} + \tfrac{1}{2}\right)\nu(\mathrm{U})\right]\mathrm{T}\right), \tag{15}$$

where $H(T)$ is defined in equation (14).

Proof. The proof of corollary 2.1 is a direct application of theorem 2.1. We denote $\tilde{\mu}$ to be the distribution of $\tilde{\boldsymbol{X}}$ defined in equation (11). Since $\tilde{\boldsymbol{X}}$ has the same distribution as $\hat{\boldsymbol{X}}$, we have, by the Hölder's inequality

$$W_p^p(\mu, \hat{\mu}) = W_p^p(\mu, \tilde{\mu}) \leqslant \mathbb{E}\left[\int_0^T \left|\boldsymbol{X}(s) - \tilde{\boldsymbol{X}}(s)\right|_2^2 \mathrm{d}s \middle| \boldsymbol{X}(0)\right]^{\frac{p}{2}}, \ 1 \leqslant p \leqslant 2. \tag{16}$$

Using equation (13) and the fact that $H(t)$ is non-decreasing w.r.t. t, we have

$$W_p(\mu, \hat{\mu}) \leqslant \sqrt{T\mathbb{E}\left[H(T)\left|\boldsymbol{X}(0)\right.\right]} \exp\left(\left[\left(\bar{f} + \tfrac{1}{2}\right) + \left(\overline{\sigma} + \tfrac{1}{2}\right) m + \left(\overline{\beta} + \tfrac{1}{2}\right)\nu(U)\right] T\right), \quad 1 \leqslant p \leqslant 2. \tag{17}$$

which proves equation (15). □

From corollary 2.1, it is necessary to have a small $W_p(\mu, \hat{\mu})$ in equation (15) such that the errors in the drift, diffusion, and jump functions $\boldsymbol{f} - \hat{\boldsymbol{f}}$, $\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}$ can be small. Note that corollary 2.1 analyzes the classic W_p-distance $W_p(\mu, \hat{\mu})$, which is different from the smooth Wasserstein distance in [26] (the classical Wasserstein distance is an upper bound for the smooth Wasserstein distance used in [28]).

$W_p(\mu, \hat{\mu}), 1 \leqslant p \leqslant 2$ cannot be directly used as a loss function to minimize as we cannot directly evaluate $\|\boldsymbol{X} - \hat{\boldsymbol{X}}\|^p$ in equation (9) since this term requires evaluation of the integral $\int_0^T \sum_{i=1}^d |X_i(t) - \hat{X}_i(t)|^2 \mathrm{d}t$. However, when $p = 2$ ($W_2(\mu, \hat{\mu})$), we shall show that we can efficiently estimate $W_2(\mu, \hat{\mu})$ by using finite-time-point observations of the two jump-diffusion processes $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$.

Let $0 = t_0 < t_1 < \ldots < t_N = T$ to be a mesh grid in the time interval $[0, T]$, and we define the following projection operator I_N

$$\boldsymbol{X}_N(t) := I_N \boldsymbol{X}(t) = \begin{cases} \boldsymbol{X}(t_i), t \in [t_i, t_{i+1}), & i < N-1, \\ \boldsymbol{X}(t_i), t \in [t_i, t_{i+1}], & i = N-1. \end{cases} \tag{18}$$

The projected $\boldsymbol{X}_N(t)$ in equation (18) is piecewise constant and is thus in the space $L^2([0, T])$. We denote the distributions of $\boldsymbol{X}_N(t)$ and $\hat{\boldsymbol{X}}_N(t) := I_N \hat{\boldsymbol{X}}_N$ in equation (18) by μ_N and $\hat{\mu}_N$, respectively. We will prove the following theorem for bounding the error $|W_2(\mu, \hat{\mu}) - W_2(\mu_N, \hat{\mu}_N)|$.

Theorem 2.2. ***(finite-time-point approximation for W_2 distance).*** *The following triangular inequality for $W_2(\mu, \hat{\mu})$ holds:*

$$W_2(\mu_N, \hat{\mu}_N) - W_2(\mu, \mu_N) - W_2(\hat{\mu}, \hat{\mu}_N) \leqslant W_2(\mu, \hat{\mu}) \leqslant W_2(\mu_N, \hat{\mu}_N) + W_2(\mu, \mu_N) + W_2(\hat{\mu}, \hat{\mu}_N). \tag{19}$$

In equation (19), $\mu_N, \hat{\mu}_N$ are the probability distributions associated with $\boldsymbol{X}_N$ and $\hat{\boldsymbol{X}}_N$ defined in equation (18), respectively. Furthermore, we assume that

$$\begin{aligned} F &:= \mathbb{E}\left[\int_0^T \sum_{i=1}^d f_i^2\left(\boldsymbol{X}(t^-), t^-\right) dt\right] < \infty, & \hat{F} &:= \mathbb{E}\left[\int_0^T \sum_{i=1}^d \hat{f}_i^2\left(\hat{\boldsymbol{X}}(t^-), t^-\right) dt\right] < \infty \\ \Sigma &:= \mathbb{E}\left[\int_0^T \sum_{\ell=1}^d \sum_{j=1}^m \sigma_{i,j}^2\left(\boldsymbol{X}(t^-), t^-\right) dt\right] < \infty, & \hat{\Sigma} &:= \mathbb{E}\left[\int_0^T \sum_{\ell=1}^d \sum_{j=1}^m \hat{\sigma}_{i,j}^2\left(\hat{\boldsymbol{X}}(t^-), t^-\right) dt\right] < \infty, \\ B &:= \mathbb{E}\left[\int_0^T \sum_{\ell=1}^d \int_U \beta_i^2\left(\boldsymbol{X}(t^-), \xi, t^-\right) \nu(d\xi)\, dt\right] < \infty, & \hat{B} &:= \mathbb{E}\left[\int_0^T \sum_{\ell=1}^d \int_U \hat{\beta}_i^2\left(\hat{\boldsymbol{X}}(t^-), \xi, t^-\right) \nu(d\xi)\, dt\right] < \infty, \end{aligned} \tag{20}$$

where $\boldsymbol{X}(t)$ and $\hat{\boldsymbol{X}}(t)$ solve equations (1) and (5), respectively. Then, we have the following bound

$$\left|W_2(\mu_N, \hat{\mu}_N) - W_2(\mu, \hat{\mu})\right| \leqslant \sqrt{\Delta t}\left(\sqrt{F\Delta t + \Sigma + B} + \sqrt{\hat{F}\Delta t + \hat{\Sigma} + \hat{B}}\right). \tag{21}$$

where $\Delta t := \max_{i=0,\ldots,N-1} |t_{i+1} - t_i|$.

The proof to theorem 2.2, given in appendix B, uses the Itô isometry as well as the orthogonal assumption of the compensated Poisson process in assumption 2.1. Theorem 2.2 indicates that $W_2(\mu, \hat{\mu})$ can be approximated by the finite-time-point projections $\boldsymbol{X}_N$ and $\hat{\boldsymbol{X}}_N$. Specifically, theorem 2.2 is a generalization to theorem 2 in [25], developed for the pure diffusion. Note that equation (21) holds for $|W_2(\mu, \hat{\mu}) - W_2(\mu_N, \hat{\mu}_N)|$ but equation (21) might not hold for $|W_p(\mu, \hat{\mu}) - W_p(\mu_N, \hat{\mu}_N)|$, $1 \leqslant p < 2$ as we cannot directly apply the Itô isometry to the compensated Poisson process for $1 \leqslant p < 2$.

It has been shown in [25] that when reconstructing pure diffusion processes, minimizing a temporally decoupled squared W_2 distance can yield more accurate reconstructions of the drift and diffusion functions than direct minimization of the squared W_2 distance $W_2^2(\mu, \hat{\mu})$ defined in equation (9). Additionally, the squared W_2 distance $W_2(\mu, \hat{\mu})$ is an upper bound of the temporally decoupled squared W_2 distance that will be discussed in section 3. Thus, corollary 2.1 also applies to the temporally decoupled squared W_2 distance.

3. A temporally decoupled squared W_2 distance

In this section, we propose and analyze a temporally decoupled squared W_2 distance, which could help effectively approximate the jump-diffusion process equation (1) by the reconstructed jump-diffusion process equation (5). Specifically, we show why this temporally decoupled squared W_2 distance can be more effectively evaluated using empirical distributions, making it a more appealing choice than the squared W_2 distance $W_2^2(\mu, \hat{\mu})$ discussed in section 2. The **temporally decoupled squared W_2 distance** is defined as

$$\tilde{W}_2^2(\mu, \hat{\mu}) := \int_0^T W_2^2(\mu(t), \hat{\mu}(t))\, \mathrm{d}t, \tag{22}$$

where $\mu(t), \hat{\mu}(t)$ are the distributions of d-dimensional random variables $\boldsymbol{X}(t)$ and $\boldsymbol{X}(t)$ at time t, respectively:

$$W_2^2(\mu(t), \hat{\mu}(t)) := \inf_{\pi(\mu(t), \hat{\mu}(t))} \mathbb{E}_{(\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t)) \sim \pi(\mu, \hat{\mu})}\left[|\boldsymbol{X}(t) - \hat{\boldsymbol{X}}(t)|_2^2\right], \tag{23}$$

where the joint distribution $\pi(\mu(t),\hat{\mu}(t))$ satisfies

$$\pi\left(\mu\left(t\right),\hat{\mu}\left(t\right)\right)\left(A,\mathbb{R}^d\right)=\mu\left(t\right)\left(A\right),\ \ \pi\left(\mu\left(t\right),\hat{\mu}\left(t\right)\right)\left(\mathbb{R}^d,B\right)=\hat{\mu}\left(t\right)\left(B\right),\ \ \forall A,B\in\mathcal{B}\left(\mathbb{R}^d\right). \tag{24}$$

The integration on the RHS of equation (22) is defined as the limit

$$\int_0^T W_2^2\left(\mu\left(t\right),\hat{\mu}\left(t\right)\right)\mathrm{d}t=\lim_{\max_i\left(t_{i+1}-t_i\right)\to 0}\sum_{i=0}^{N-1}W_2^2\left(\mu\left(t_i\right),\hat{\mu}\left(t_i\right)\right)\Delta t_i, \tag{25}$$

where $0=t_0<t_1<\ldots<t_N=T$ is a grid mesh on the time interval $[0,T]$ and $\Delta t_i:=t_{i+1}-t_i$ in the following. Here, we shall prove that the temporally decoupled squared W_2 distance $\tilde{W}_2^2(\mu,\hat{\mu})$ is well defined and can be a more effective loss function when seeking to reconstruct multidimensional jump-diffusion processes than the original squared $W_2^2(\mu_N,\hat{\mu}_N)$. Two features make this so: i) numerically evaluating the temporally decoupled squared W_2 distance equation (22) using finite-sample empirical distributions can be more accurate than evaluating the original squared W_2 distance $W_2^2(\mu,\hat{\mu})$ when the number of training samples becomes larger, and ii) the temporally decoupled squared W_2 distance equation (22) provides upper error bounds for $\boldsymbol{f}-\hat{\boldsymbol{f}},\boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta}-\hat{\boldsymbol{\beta}}$ when reconstructing jump-diffusion processes.

We denote $\boldsymbol{\mu}_i$ and $\hat{\boldsymbol{\mu}}_i$ to be the distributions for $\boldsymbol{X}(t),t\in[t_i,t_{i+1})$ and $\hat{\boldsymbol{X}}(t),t\in[t_i,t_{i+1})$, respectively. We can prove the following theorem that shows the limit on the RHS of equation (25) exists and thus the temporally decoupled squared W_2 in equation (22) distance is well defined.

Theorem 3.1 ***(the temporally decoupled squared W_2 distance is well-defined).*** *We assume the conditions in theorem 2.2 hold. Furthermore, we assume that for any $0<t<t'<T$, as $t'-t\to 0$, the following conditions are satisfied*

$$\begin{aligned}
&\mathbb{E}\left[\int_t^{t'}\sum_{i=1}^{d}f_i^2\left(\boldsymbol{X}(t),t\right)dt\right]\to 0, && \mathbb{E}\left[\int_t^{t'}\sum_{i=1}^{d}\hat{f}_i^2\left(\hat{\boldsymbol{X}}(s^-),s^-\right)ds\right]\to 0,\\
&\mathbb{E}\left[\int_t^{t'}\sum_{\ell=1}^{d}\sum_{j=1}^{m}\sigma_{i,j}^2\left(\boldsymbol{X}(s^-),s^-\right)ds\right]\to 0, && \mathbb{E}\left[\int_t^{t'}\sum_{\ell=1}^{d}\sum_{j=1}^{m}\hat{\sigma}_{i,j}^2\left(\hat{\boldsymbol{X}}(s^-),s^-\right)ds^-\right]\to 0,\\
&\mathbb{E}\left[\int_t^{t'}\sum_{\ell=1}^{d}\int_U\beta_\ell^2\left(\boldsymbol{X}(s^-),\xi,s^-\right)\nu\left(d\xi\right)ds\right]\to 0, && \mathbb{E}\left[\int_t^{t'}\sum_{\ell=1}^{d}\int_U\hat{\beta}_\ell^2\left(\hat{\boldsymbol{X}}(s^-),\xi,s^-\right)\nu\left(d\xi\right)ds\right]\to 0.
\end{aligned} \tag{26}$$

Additionally, we assume that there is a uniform upper bound

$$M:=\max_{t\in[0,T]}W_2\left(\mu\left(t\right),\hat{\mu}\left(t\right)\right)\leqslant\infty. \tag{27}$$

Suppose $\Delta t:=\max_{0\leqslant i\leqslant N-1}\Delta t_i$, then

$$\lim_{\Delta t\to 0}\left(\sum_{i=0}^{N-1}W_2^2\left(\mu(t_i),\hat{\mu}(t_i)\right)\Delta t_i-\sum_{i=0}^{N-1}W_2\left(\boldsymbol{\mu}_i,\hat{\boldsymbol{\mu}}_i\right)\right)=0. \tag{28}$$

Furthermore, the limit

$$\lim_{\Delta t\to 0}\sum_{i=0}^{N-1}W_2^2\left(\mu\left(t_i\right),\hat{\mu}\left(t_i\right)\right)\Delta t_i=\lim_{N\to\infty}\sum_{i=0}^{N-1}W_2^2\left(\mu\left(t_i\right),\hat{\mu}\left(t_i\right)\right)\Delta t_i \tag{29}$$

is simply $\tilde{W}_2^2(\mu,\hat{\mu})$ defined in equation (22). Denoting π_i to be the coupling probability distribution of $(\boldsymbol{X}(t_i),\hat{\boldsymbol{X}}(t_i))$, whose marginal distributions coincide with $\mu(t_i)$ and $\hat{\mu}(t_i)$, we have the following bound:

$$\left|\sum_{i=0}^{N-1}\inf_{\pi_i}\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}(t_i)-\hat{\boldsymbol{X}}(t_i)\right|_2^2\right]\Delta t_i-\tilde{W}_2^2\left(\mu,\hat{\mu}\right)\right|\leqslant 2MT\max_i\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right), \tag{30}$$

where

$$
\begin{aligned}
F_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{i=1}^{d} f_i^2\left(\boldsymbol{X}(t^-), t^-\right) dt\right], \qquad & \hat{F}_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{i=1}^{d} \hat{f}_i^2\left(\hat{\boldsymbol{X}}(t^-), t^-\right) dt\right], \\
\Sigma_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{\ell=1}^{d}\sum_{j=1}^{m} \sigma_{\ell,j}^2\left(\boldsymbol{X}(t^-), t^-\right) dt\right], \qquad & \hat{\Sigma}_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{\ell=1}^{d}\sum_{j=1}^{m} \hat{\sigma}_{\ell,j}^2\left(\hat{\boldsymbol{X}}(t^-), t^-\right) dt\right], \\
B_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{\ell=1}^{d} \int_U \beta_\ell^2\left(\boldsymbol{X}(t^-), \xi, t^-\right) \nu\left(d\xi\right) dt\right], \qquad & \hat{B}_i :=& \mathbb{E}\left[\int_{t_i}^{t_{i+1}} \sum_{\ell=1}^{d} \int_U \hat{\beta}_\ell^2\left(\hat{\boldsymbol{X}}(t^-), \xi, t^-\right) \nu\left(d\xi\right) dt\right].
\end{aligned} \tag{31}
$$

Theorem 3.1 generalizes theorem 3 in [25] from pure diffusion processes to jump-diffusion processes. The proof to theorem 3.1 is in appendix C. Specifically, from equation (30), if $\max_i\left(F_i\Delta t + \Sigma_i + B_i + \hat{F}_i\Delta t + \hat{\Sigma}_i + \hat{B}_i\right)$ is of order Δt, then the convergence rate of $\sum_{i=1}^{N-1} \inf_{\pi_i}\ \mathbb{E}_{\pi_i}[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2]\Delta t_i$ to $\tilde{W}_2^2(\mu, \hat{\mu})$ is $O(\sqrt{\Delta t})$. Specifically, we have

$$
\left|W_2^2\left(\mu, \hat{\mu}\right) - W_2^2\left(\mu_N, \hat{\mu}_N\right)\right| = \left|W_2\left(\mu, \hat{\mu}\right) - W_2\left(\mu_N, \hat{\mu}_N\right)\right| \cdot \left|W_2\left(\mu, \hat{\mu}\right) + W_2\left(\mu_N, \hat{\mu}_N\right)\right|. \tag{32}
$$

From equation (21), the error bound of $|W_2(\mu, \hat{\mu}) - W_2(\mu_N, \hat{\mu}_N)|$ is $O\left(\sqrt{\Delta t}\right)$. Therefore, the upper error bounds of using the finite-time distributions $\mu_N, \hat{\mu}_N$ to approximate both $W_2^2(\mu, \hat{\mu})$ or $\tilde{W}_2^2(\mu, \hat{\mu})$ are both of order $O(\sqrt{\Delta t})$.

Next, we shall show that using the finite-sample empirical distribution to estimate

$$
\sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i, \tag{33}
$$

where π_i is the coupling distribution of $(\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i))$ such that its marginal distributions are $\mu(t_i)$ and $\hat{\mu}(t_i)$, is more accurate than using the finite-sample empirical distribution to estimate $W_2^2(\mu_N, \hat{\mu}_N)$ (where μ_N and $\hat{\mu}_N$ are the distributions of $\boldsymbol{X}_N(t)$ and $\hat{\boldsymbol{X}}_N(t)$ defined in equation (18)).

Theorem 3.2 (finite sample empirical distribution error bound). *We assume that*

$$
\mathbb{E}\left[|\boldsymbol{X}(t)|_6^6\right] \leqslant \infty,\ \mathbb{E}\left[|\hat{\boldsymbol{X}}(t)|_6^6\right] \leqslant \infty,\ \forall t \in [0, T], \tag{34}
$$

where $|\cdot|_6$ is the l^6 norm of a vector in $\mathbb{R}^d$. We denote $\mu_N^e, \hat{\mu}_N^e$ to be empirical distributions of $\boldsymbol{X}_N$ and $\hat{\boldsymbol{X}}_N$, respectively; we denote $\mu_N^e(t_i), \hat{\mu}_N^e(t_i)$ to be the empirical distributions of $\boldsymbol{X}(t_i)$ and $\hat{\boldsymbol{X}}(t_i), i = 0, 1, \ldots, N-1$. Suppose M_s is the number of observed trajectories $\boldsymbol{X}_N(t_i)$ and the number of reconstructed trajectories $\hat{\boldsymbol{X}}_N(t_i)$. We find the following error bound for estimating $W_2^2(\mu_N, \hat{\mu}_N)$ using the empirical distributions:

$$
\begin{aligned}
&\mathbb{E}\left[\left|W_2^2\left(\mu_N^e, \hat{\mu}_N^e\right) - W_2^2\left(\mu_N, \hat{\mu}_N\right)\right|\right] \leqslant E_1\left(M_s\right), \quad \textit{where} \\
&E_1\left(M\right) := 2\sqrt{C_0} W_2\left(\mu_N, \hat{\mu}_N\right) h\left(M_s, Nd\right) \sum_{i=0}^{N-1} \left(\mathbb{E}\left[|\boldsymbol{X}(t_i)|_6^6\right]^{\frac{1}{6}} + \mathbb{E}\left[|\hat{\boldsymbol{X}}(t_i)|_6^6\right]^{\frac{1}{6}}\right) \sqrt{\Delta t_i} \\
&\qquad + 2C_0 h^2\left(M_s, Nd\right) \sum_{i=0}^{N-1} \left(\mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{3}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{3}}\right) \Delta t_i,
\end{aligned} \tag{35}
$$

where C_0 is a constant and

$$
h\left(M_s, n\right) := \begin{cases} M_s^{-\frac{1}{4}} \log\left(1 + M_s\right)^{\frac{1}{2}}, & n \leqslant 4, \\ M_s^{-\frac{1}{n}}, & n > 4 \end{cases} \tag{36}
$$

We also have the empirical error bound for estimating $\sum_{i=1}^{N-1} W_2^2(\mu_N(t_i),\mu_N(t_i))\Delta t_i$ *using the empirical distributions* $\sum_{i=1}^{N-1} W_2^2(\mu_N^e(t_i),\mu_N^e(t_i))\Delta t_i$*:*

$$\begin{aligned}
&\mathbb{E}\left[\left|\sum_{i=0}^{N-1} W_2^2\left(\mu_N^e(t_i),\mu_N^e(t_i)\right)\Delta t_i - W_2^2\left(\mu_N(t_i),\mu_N(t_i)\right)\Delta t_i\right|\right] \\
&\quad\leqslant \mathbb{E}\left[\sum_{i=0}^{N-1}\left|W_2^2\left(\mu_N^e(t_i),\mu_N^e(t_i)\right) - W_2^2\left(\mu_N(t_i),\mu_N(t_i)\right)\right|\Delta t_i\right] \leqslant E_2(M_s), \quad \textit{where} \\
&E_2(M_s) := 2\sqrt{C_1}h(M_s,d)\sum_{i=0}^{N-1}\left(\mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{6}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{6}}\right)\Delta t_i W_2\left(\mu_N(t_i),\hat{\mu}_N(t_i)\right) \\
&\qquad + 2C_1h^2(M_s,d)\sum_{i=0}^{N-1}\left(\mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{3}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{3}}\right)\Delta t_i,
\end{aligned} \tag{37}$$

where C_1 *is a constant different from* C_0*. Furthermore, there exists a constant* C *such that*

$$E_1(M_s) \geqslant CE_2(M_s)\cdot\frac{h(M_s,Nd)}{h(M_s,d)}N^{-\frac{2}{3}}. \tag{38}$$

The proof of theorem 3.2 is given in appendix D and utilizes the upper bound of the W-distance between the ground truth distribution and the empirical distribution in [29]. Specifically, if $N \geqslant 5$, then $h(M_s,Nd) = 2M_s^{-\frac{1}{Nd}}$, and

$$\frac{h(M_s,Nd)}{h(M_s,d)} \geqslant \min\left\{M_s^{\frac{1}{4}-\frac{1}{Nd}}\log(1+M_s), M_s^{\frac{N-1}{Nd}}\right\}. \tag{39}$$

Therefore, theorem 3.2 indicates that as the number of observed trajectories of the jump-diffusion process M_s increases, the upper bound of $\mathbb{E}\left[\left|\sum_{i=0}^{N-1} W_2^2(\mu_N^e(t_i),\mu_N^e(t_i))\Delta t_i - \sum_{i=0}^{N-1} W_2^2\left(\mu_N(t_i),\mu_N(t_i)\right)\Delta t_i\right|\right]$ converges faster to 0 than the upper bound of $\mathbb{E}\left[\left|W_2^2(\mu_N^e,\hat{\mu}_N^e) - W_2^2(\mu_N,\hat{\mu}_N)\right|\right]$ does when $N \geqslant 5$. Thus,

$$\sum_{i=0}^{N-1} W_2^2\left(\mu_N(t_i),\mu_N(t_i)\right)\Delta t_i \tag{40}$$

can be more accurately evaluated by the finite-sample empirical distributions than the squared W_2 distance $W_2^2(\mu_N,\hat{\mu}_N)$ when M_s is large.

For any coupled distribution $\pi(\boldsymbol{X}_N,\hat{\boldsymbol{X}}_N)$ such that its marginal distributions are μ_N and $\hat{\mu_N}$, its marginal distributions w.r.t $\boldsymbol{X}(t_i)$ and $\hat{\boldsymbol{X}}(t_i)$ are $\mu(t_i)$ and $\hat{\mu}(t_i)$, respectively. Thus,

$$\sum_{i=0}^{N-1}\inf_{\pi_i}\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}(t_i)-\hat{\boldsymbol{X}}(t_i)\right|_2^2\right]\Delta t_i \leqslant \inf_{\pi\left(\boldsymbol{X}_N,\hat{\boldsymbol{X}}_N\right)}\sum_{i=0}^{N-1}\mathbb{E}_{\pi\left(\boldsymbol{X}_N,\hat{\boldsymbol{X}}_N\right)}\left[\left|\boldsymbol{X}(t_i)-\hat{\boldsymbol{X}}(t_i)\right|_2^2\right]\Delta t_i = W_2^2(\mu_N,\hat{\mu}_N). \tag{41}$$

Letting $N\to\infty$ in equation (41), from theorems 2.2 and 3.1, we conclude that

$$\tilde{W}_2^2(\mu,\hat{\mu}) \leqslant W_2^2(\mu,\hat{\mu}). \tag{42}$$

Thus, corollary 2.1 also provides an upper error bound for the temporally decoupled $\tilde{W}_2^2(\mu,\hat{\mu})$. Next, we show that there is a lower bound for $\tilde{W}_2^2(\mu,\hat{\mu})$ and this lower bound depends on drift, diffusion, and jump functions of the ground truth jump-diffusion process equation (1) and the approximate jump-diffusion process equation (5).

Theorem 3.3 (lower error bound for the temporally decoupled squared W_2-distance). *We have the following lower bound:*

$$\tilde{W}_2^2(\mu,\hat{\mu}) \geqslant \int_0^T\sum_{i=1}^d\left(\mathbb{E}\left[\int_0^t\left[f_i\left(\boldsymbol{X}(s^-),s^-\right)-\hat{f}_i\left(\hat{\boldsymbol{X}}(s^-),s^-\right)\right]ds\right]\right)^2 dt + \int_0^T Tr\left(\boldsymbol{S}_t+\hat{\boldsymbol{S}}_t-2\left(\boldsymbol{S}_t\hat{\boldsymbol{S}}_t\right)^{\frac{1}{2}}\right)dt, \tag{43}$$

where $\boldsymbol{S}_t, \hat{\boldsymbol{S}}_t$ are two matrices in $\mathbb{R}^{d\times d}$ with their elements defined by

$$\begin{aligned}(\boldsymbol{S}_t)_{i,j} :=& \mathbb{E}\left[\sum_{\ell=1}^{m}\int_0^t \sigma_{i,\ell}\left(\boldsymbol{X}(s^-),s^-\right)\cdot\sigma_{j,\ell}\left(\boldsymbol{X}(s^-),s^-\right)ds\right]\\ &+\mathbb{E}\left[\int_0^t\int_U \beta_i\left(\boldsymbol{X}(s^-),\xi,s^-\right)\cdot\beta_j\left(\boldsymbol{X}(s^-),\xi,s^-\right)\nu\left(d\xi\right)ds\right],\\ \left(\hat{\boldsymbol{S}}_t\right)_{i,j} :=& \mathbb{E}\left[\sum_{\ell=1}^{m}\int_0^t \hat{\sigma}_{i,\ell}\left(\hat{\boldsymbol{X}}(s^-),s^-\right)\cdot\hat{\sigma}_{j,\ell}\left(\hat{\boldsymbol{X}}(s^-),s^-\right)ds\right]\\ &+\mathbb{E}\left[\int_0^t\int_U \beta_i\left(\boldsymbol{X}(s^-),\xi,s^-\right)\cdot\beta_j\left(\boldsymbol{X}(s^-),\xi,s^-\right)\nu\left(d\xi\right)ds\right].\end{aligned} \tag{44}$$

The terms $(\boldsymbol{S}_t)^{\frac{1}{2}}$ and $(\hat{\boldsymbol{S}}_t)^{\frac{1}{2}}$ indicate the positive square-roots.

Proof. First, we denote

$$\boldsymbol{X}_0(t) := \boldsymbol{X}(t) - \mathbb{E}\left[\boldsymbol{X}(t)\right],\ \hat{\boldsymbol{X}}_0(t) := \hat{\boldsymbol{X}}(t) - \mathbb{E}\left[\hat{\boldsymbol{X}}(t)\right] \tag{45}$$

and let $\mu_0(t)$ and $\hat{\mu}_0(t)$ to be the probability distributions of $\boldsymbol{X}_0(t)$ and $\hat{\boldsymbol{X}}_0(t)$, respectively. From theorem 1 in [30], we have

$$W_2^2\left(\mu_0(t),\hat{\mu}_0(t)\right) \geqslant \mathrm{Tr}\left(\boldsymbol{S}_t+\hat{\boldsymbol{S}}_t-2\left(\boldsymbol{S}_t\hat{\boldsymbol{S}}_t\right)^{\frac{1}{2}}\right). \tag{46}$$

Because

$$W_2^2\left(\mu(t),\hat{\mu}(t)\right) = W_2^2\left(\mu_0(t),\hat{\mu}_0(t)\right) + \sum_{i=1}^{d}\left(\mathbb{E}\left[\int_0^t\left(f_i\left(\boldsymbol{X}(s^-),s^-\right)-\hat{f}_i\left(\hat{\boldsymbol{X}}(s^-),s^-\right)\right)\mathrm{d}s\right]\right)^2, \tag{47}$$

equation (43) holds, proving theorem 3.3. □

Theorem 3.3 gives a lower bound for the temporally decoupled $\tilde{W}(\mu,\hat{\mu})$. Specifically, if $d=1$, equation (43) can be further simplified to

$$\begin{aligned}\tilde{W}_2^2(\mu,\hat{\mu}) \geqslant& \int_0^T\left(\mathbb{E}\left[\int_0^t f_1\left(X(s^-),s^-\right)\mathrm{d}s\right]-\mathbb{E}\left[\int_0^t \hat{f}_1\left(\hat{X}(s^-),s^-\right)\mathrm{d}s\right]\right)^2\mathrm{d}s\\ &+\int_0^T\left(\mathbb{E}\left[\int_0^t\sigma^2\left(X(s^-),s^-\right)\mathrm{d}s+\int_U\beta^2\left(X(s^-),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]^{\frac{1}{2}}\right.\\ &\left.-\mathbb{E}\left[\int_0^t\hat{\sigma}^2\left(\hat{X}(s^-),s^-\right)\mathrm{d}s+\int_U\hat{\beta}^2\left(\hat{X}(s^-),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]^{\frac{1}{2}}\right)^2\mathrm{d}t.\end{aligned} \tag{48}$$

Thus, if the jump-diffusion process to be reconstructed is one-dimensional (equation (1)), then we conclude that, as $\tilde{W}_2^2(\mu,\hat{\mu})\to 0$,

$$\begin{aligned}&\mathbb{E}\left[\int_0^t f_1\left(X_1(s^-),s^-\right)\mathrm{d}s\right]-\mathbb{E}\left[\int_0^t\hat{f}_1\left(\hat{X}(s^-),s^-\right)\mathrm{d}s\right]\to 0,\ \text{a.s.}\quad\text{and}\\ &\mathbb{E}\left[\int_0^t\sigma^2\left(X_1(s^-),s^-\right)\mathrm{d}s+\int_U\beta^2\left(X_1(s^-),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]^{\frac{1}{2}}\\ &\quad-\mathbb{E}\left[\int_0^t\hat{\sigma}^2\left(\hat{X}_1(s^-),s^-\right)\mathrm{d}s+\int_U\hat{\beta}^2\left(\hat{X}_1(s^-),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]^{\frac{1}{2}}\to 0,\ \text{a.s.}\end{aligned} \tag{49}$$

However, equation (49) does not imply that either

$$\mathbb{E}\left[\int_0^t\hat{\sigma}^2\left(\hat{X}_1(s^-),s^-\right)\mathrm{d}s\right]-\mathbb{E}\left[\int_0^t\hat{\sigma}^2\left(\hat{X}_1(s^-),s^-\right)\mathrm{d}s\right]\to 0 \tag{50}$$

or

$$\mathbb{E}\left[\int_0^t\int_U\hat{\beta}^2\left(X_1\left(s^-\right),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]-\mathbb{E}\left[\int_0^t\int_U\hat{\beta}^2\left(X_1\left(s^-\right),\xi,s^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}s\right]\to 0. \tag{51}$$

A lower bound for the temporally decoupled squared W_2 distance between two jump-diffusion processes that depends on the expectation of the summation of the error in the jump and the error in the diffusion functions is worth further investigation. Such an intricate analysis is beyond the scope of this paper but could imply that minimizing the W_2 distance is necessary for a good reconstruction of both the diffusion function and the jump function. Moreover, when $d > 1$, it is not easy to make further simplifications to equation (43). Analysis of the properties of the matrix $\boldsymbol{S}_t + \hat{\boldsymbol{S}}_t - 2(\boldsymbol{S}_t\hat{\boldsymbol{S}}_t)^{\frac{1}{2}}$ in equation (43) can be quite difficult. Nonetheless, we shall show in our numerical examples that our temporally decoupled squared W_2-distance $\tilde{W}(\mu, \hat{\mu})$ method can accurately reconstruct both the diffusion and the jump functions in several examples of one-dimensional and multidimensional jump-diffusion processes especially when the drift function can be provided as prior information.

4. Numerical experiments

In this section, we implement our methods through numerical examples and investigate the effectiveness of the temporally decoupled squared W_2-distance method in the reconstruction the jump-diffusion process equation (1). We also compare our results with those derived from using other commonly used losses in uncertainty quantification and methods for jump-diffusion process reconstruction. Additionally, we explore how prior knowledge on the ground truth jump-diffusion process equation (1) helps in its reconstruction. All experiments are carried out using Python 3.11 on a desktop with a 32-core Intel® i9-13 900KF CPU (when comparing runtimes, we train each model on just one core).

In all experiments, we use three feed-forward neural networks to parameterize the drift, diffusion, and jump functions in the approximate jump-diffusion process equation (5), i.e.

$$\hat{f} := \hat{f}(X, t; \Theta_1)\,,\ \hat{\sigma} := \hat{\sigma}(X, t; \Theta_2)\,,\ \hat{\beta} := \hat{\sigma}(X, \xi, t; \Theta_3)\,. \tag{52}$$

$\Theta_1, \Theta_2, \Theta_3$ are the parameter sets in the three parameterized neural networks, respectively. We modified the `torchsde` Python package in [18] to implement the Euler-Maruyama scheme for generating trajectories of the two jump-diffusion processes equations (1) and (5). Details of the training settings and hyperparameters for all examples are given in appendix E. In examples 4.1 and 4.2, the reconstruction errors are the relative L^2 errors:

$$\text{drift error} := \frac{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\left|f\left(x_j(t_i), t_i\right) - \hat{f}\left(x_j(t_i), t_i\right)\right|}{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\left|f\left(x_j(t_i), t_i\right)\right|}, \tag{53}$$

$$\text{diffusion error} := \frac{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\left|\left|\sigma\left(x_j(t_i), t_i\right)\right| - \left|\hat{\sigma}\left(x_j(t_i), t_i\right)\right|\right|}{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\left|\sigma\left(x_j(t_i), t_i\right)\right|} \tag{54}$$

$$\text{jump error} := \frac{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\int_U\left|\beta\left(x_j(t_i), \xi, t_i\right) - \hat{\beta}\left(x_j(t_i), \xi, t_i\right)\right|\mathrm{d}\nu(\xi)}{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\int_U\left|\beta\left(x_j(t_i), t_i\right)\right|\mathrm{d}\nu(\xi)}, \tag{55}$$

where N is the number of time steps and M_s is the number of training trajectories.

Example 4.1. For our first example, we reconstruct the following 1D jump-diffusion process for describing the non-defaultable zero-coupon bond pricing [2]:

$$\mathrm{d}X_t = (b + aX_t)\,\mathrm{d}t + \sigma_0\sqrt{|X_t|}\mathrm{d}B_t + \mathrm{d}C_t,\ \ a, b, \sigma_0 \in \mathbb{R}, t \in [0, T] \tag{56}$$

where $C_t = \sum_{i=1}^{N_t} Y_i$, Y_i are independently identically distributed, and N_t obeys the Poisson distribution with intensity t. We take $Y_i \equiv y_0$ so that equation (56) can be rewritten as

$$\mathrm{d}X_t = (b + y_0 + aX_t)\,\mathrm{d}t + \sigma_0\sqrt{|X_t|}\mathrm{d}B_t + y_0\mathrm{d}\tilde{N}_t,\ \ t \in [0, T]\,, \tag{57}$$

with $\tilde{N}_t$ a 1D compensated Poisson process with intensity t. We define ground truth by $b = 4, a = -1, \sigma_0 = 0.4, y_0 = 1$ in equation (57) and take $T = 20.2$ and initial condition $X_0 = 2$. We reconstruct equation (57) by minimizing the temporally decoupled W-distance equation (40).

We compare our temporally decoupled squared W_2 distance loss function with the WGAN method and other loss functions (MSE, MMD, mean^2+var, W_1 distance, and the squared W_2 distance $W_2^2(\mu_N, \hat{\mu}_N)$. The definitions of the other loss functions are given in appendix F). As shown in figures 1(a)–(f), the trajectories we obtained by minimizing our temporally decoupled squared W_2-distance accurately match the ground truth trajectories generated by equation (57). When using $W_1(\mu, \hat{\mu})$, $W_2^2(\mu, \hat{\mu})$, and MSE loss functions, the

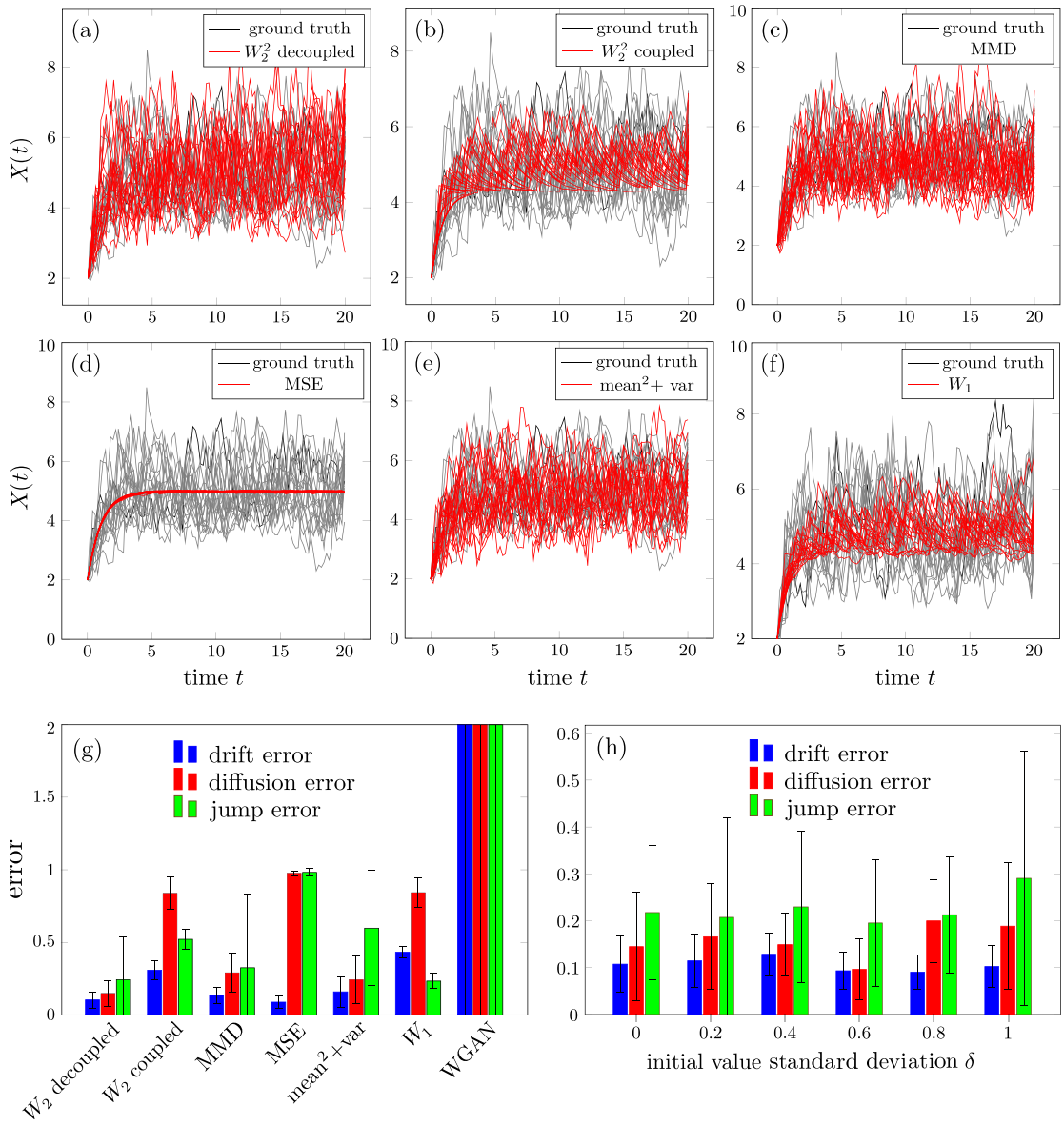

Figure 1. Reconstruction of trajectories and model functions. We define ground truth as $b = 4, a = -1, \sigma_0 = 0.4, y_0 = 1$ in equation (57), with $T = 20.2$ and initial condition $X_0 = 2$. (a)–(f) ground truth (black) and reconstructed trajectories (red) generated from the learned jump-diffusion process by minimizing different loss functions or using different methods. (g) The reconstruction errors of the drift, diffusion, and jump functions defined in equations (53)–(55). We compare errors from minimizing our temporally decoupled squared W_2-distance versus those from minimizing the MSE, MMD, mean^2+var, the W_1-distance $W_1(\mu, \hat{\mu})$, the squared W_2-distance $W_2^2(\mu_N, \hat{\mu}_N)$, and the error of results obtained using the WGAN method. The mean and standard deviation of the error for different methods are obtained by repeating the experiment 10 times. (h) The reconstruction errors in the drift, diffusion, and jump functions defined in equations (53)–(55) w.r.t. the standard deviation δ of the initial condition (equation (58)).

reconstructed trajectories deviate qualitatively from those of the ground truth. The solutions of the reconstructed jump-diffusion process generated by the WGAN method are also qualitatively incorrect and are thus not shown here. From figure 1(g), minimizing our temporally decoupled squared W_2-distance gives the smallest reconstruction errors $f - \hat{f}, \sigma - \hat{\sigma}$, and $\beta - \hat{\beta}$. The average errors in the reconstructed drift, diffusion, and jump are kept below 0.25. Thus, minimizing our temporally decoupled squared W_2 distance is found to be more accurate in reconstructing the jump-diffusion process equation (57) than other benchmark methods. We also list the average runtime per training iteration as well as the memory usage of different methods in table 1. The runtime of the WGAN method is significantly longer than that of other methods. Furthermore, the computational cost of using our temporally decoupled squared W_2 is similar to the cost of using other loss functions while our temporally decoupled squared W_2 method can accurately reconstruct equation (57).

We also evaluate the numerical performance of different loss functions as we vary the number of training trajectories sampled from the ground truth jump-diffusion process in equation (56). The reconstruction accuracy of the drift, diffusion, and jump functions for all methods tends to improve with an increased

Table 1. The runtime and memory usage of different methods (loss functions) when reconstructing the jump-diffusion process equation (57).

Method (loss)	temporally decoupled W_2^2	W_2^2	MMD	MSE	mean^2+var	W_1	WGAN
Average time/iteration (s)	44.5	48.2	59.4	29.8	48.7	39.2	346.3
Average memory use (Gb)	2.61	3.52	2.59	5.00	2.61	2.60	3.34

number of trajectories for training. Additionally, our proposed temporally decoupled squared W_2 method gives more accurate reconstructed drift, diffusion, and jump functions than most other loss functions or methods. Minimizing the MMD loss function can yield an even more accurate reconstructed jump function when the number of training trajectories is small; however, using the MMD as the loss function gives a less accurate reconstruction of the drift and diffusion functions than our proposed temporally decoupled squared W_2 method. Results are given in appendix G.

Additionally, we test the numerical performance of our temporally decoupled squared W_2 method when reconstructing equation (57) under different initial conditions. Instead of using the same initial condition for all solutions, we sample the initial value from

$$X_0 \sim \mathcal{N}\left(2, \delta^2\right), \tag{58}$$

where $\mathcal{N}(2,\delta^2)$ is the 1D normal distribution of mean 2 and variance δ^2. Using the same hyperparameters in the neural networks and for training (in table appendix E) as in example 4.1, we varied the standard deviation $\delta = 0, 0.2, 0.4, 0.6, 0.8, 1$ and implemented the temporally decoupled squared W_2 distance as a loss function. The results shown in figure 1(h) indicate that the reconstruction of equation (57) using the squared W_2 loss function is rather insensitive to 'noise', i.e. the standard deviation δ in the distribution of the initial condition.

Finally, we also use different values of the parameters σ_0 and y_0 in the diffusion and drift functions. The reconstructed drift functions $\hat{f}$ remain accurate when σ_0 and y_0 are varied. When σ_0, y_0 are small, the corresponding diffusion and jump functions can also be accurately reconstructed; however, when σ_0, y_0 are large, the reconstruction of the diffusion function can be less accurate because the trajectories for training are more sparsely distributed. Details of the results are given in appendix H.

It was shown in [25] that the accuracy of reconstructing a pure-diffusion process ($\boldsymbol{\beta} = 0$ in equation (1)) can deteriorate if trajectories for training are too sparsely distributed (too few trajectories/too high noise). However, we find that prior information on the drift function in equation (1) enables efficient reconstruction even if the number of training trajectories is limited, when the temporally decoupled squared W_2 method for reconstructing jump-diffusion processes without prior information would otherwise fail. In the next example, we demonstrate enhanced reconstruction performance after incorporating prior information on the drift function of equation (1), greatly improving the accuracy of reconstructed diffusion and jump functions.

Example 4.2. Consider the following 1D jump-diffusion process

$$\mathrm{d}X_t = \alpha\left(X_t, t\right)\mathrm{d}t + \sigma\left(X_t, t\right)\mathrm{d}B_t + \beta\left(X_t, t\right)\mathrm{d}\tilde{N}_t, \quad t \in [0, T], \tag{59}$$

where $\tilde{N}_t$ is a 1D compensated Poisson process with intensity t. This model, if we set $S_t \equiv e^{X_t}$, can describe the posited stock returns under a deterministic jump ratio [1]. To test the efficiency of our temporally decoupled squared W_2-distance method, we set $\alpha \equiv r_0 = 0.05$ (i.e. the drift function to be a constant risk-free interest rate [31]), the initial condition $X_0 = 1$, and $T = 5.1$ and explore different forms of the diffusion and jump functions $\sigma(X,t)$ and $\beta(X,t)$. We then input the drift, diffusion, or the jump function α, σ, or β in equation (59) as prior information to test how well our method can reconstruct the other terms.

Summarizing, (i) we first give no prior information and reconstruct all three functions α, σ, and β; (ii) we specify the risk-free interest rate $\alpha \equiv r_0$ and reconstruct σ, and β; (iii) we provide the diffusion function σ and reconstruct α and β; (iv) we provide the jump function β and reconstruct f, and σ. In this example, 'const' refers to using a constant diffusion or jump function:

$$\sigma\left(X,t\right) \equiv \sigma_0 \text{ or } \beta\left(X,t\right) \equiv \beta_0, \tag{60}$$

'linear' refers to using a linear diffusion or jump function

$$\sigma\left(X,t\right) \equiv \sigma_0 X \text{ or } \beta\left(X,t\right) \equiv \beta_0 X, \tag{61}$$

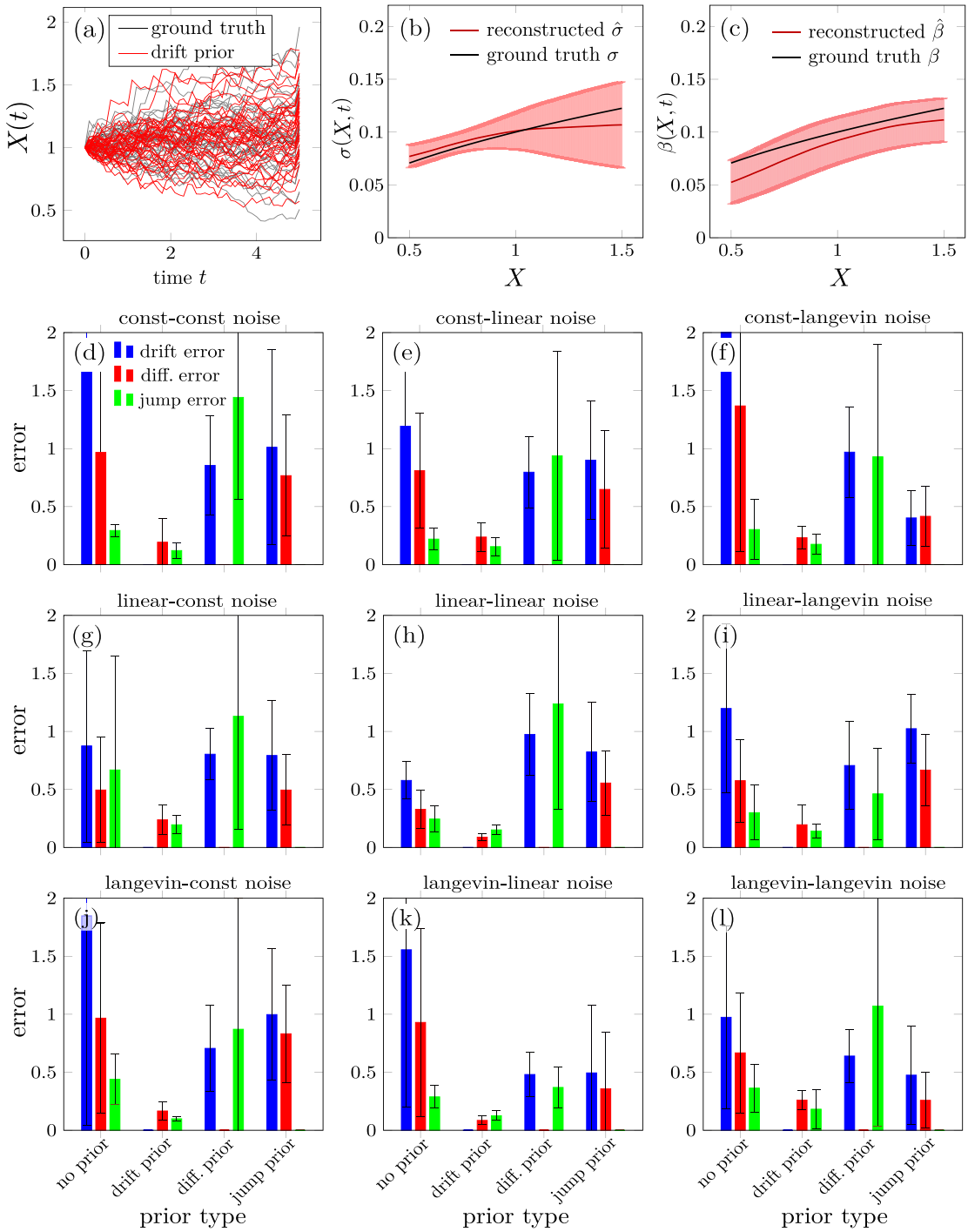

Figure 2. (a) The trajectories generated by the ground truth (black) jump-diffusion process with $\sigma(X,t) \equiv 0.1\sqrt{|X|}$ and $\beta(X,t) \equiv 0.1\sqrt{|X|}$ and given drift function in equation (59), plotted against reconstructed trajectories (red) using the same drift function prior. (b) and (c) The ground truth diffusion and jump functions $\sigma(X,t) \equiv \sigma_0\sqrt{|X|}$ and $\beta(X,t) \equiv \beta_0\sqrt{|X|}$) shown against the reconstructed functions $\hat{\sigma}(X,t)$ and $\hat{\beta}(X,t)$. (with drift function given as prior). The red curves are the mean $\hat{\sigma}(X,t)$ and $\hat{\beta}(X,t)$ while the shaded bands show their standard deviations, calculated over 5 independent experiments). (d)–(k) The reconstruction errors of the drift, diffusion, and jump functions without prior information on equation (59) or with one of the drift, diffusion, and jump functions given. When the drift function is given, errors in the reconstructed diffusion and jump functions are the smallest in all cases (error bars under 'drift prior.').

and 'langevin' refers to using a diffusion or jump function of the following form

$$\sigma\,(X,t) \equiv \sigma_0\sqrt{|X|} \text{ or } \beta\,(X,t) \equiv \beta_0\sqrt{|X|}. \tag{62}$$

To illustrate the reconstruction, we set $\sigma_0 = \beta_0 = 0.1$ in equations (60)–(62), and plot in figure 2(a) the ground truth solutions (black) generated from equation (59) with a given drift function. Using the same drift function, trajectories of the reconstructed jump-diffusion process are shown in red and exhibit a distribution

Table 2. Average errors in the reconstructed drift, diffusion, and jump functions when using the temporally decoupled squared W_2 distance to reconstruct equation (59). The error is taken over 9 possible combinations of different forms of diffusion and jump functions (constant, Langevin, and linear in equations (60)–(62)) in figure 2.

Prior info	error of reconstructed $\hat{f}$	error of reconstructed $\hat{\sigma}$	error of reconstructed $\hat{\beta}$
No prior	1.412($\pm$1.520)	0.790($\pm$0.714)	0.347($\pm$0.356)
Given $\alpha(x,t)$	0	0.189($\pm$0.124)	0.150($\pm$0.080)
Given $\sigma(x,t)$	0.771($\pm$0.333)	0	0.939($\pm$0.854)
Given $\beta(x,t)$	0.769($\pm$0.520)	0.556($\pm$0.393)	0

that matches well with that of the ground truth solutions. Moreover, as shown in figures 2(b) and (c), the differences between the learned diffusion and jump functions $\hat{\sigma}(X,t)$ and $\hat{\beta}(X,t)$ and the ground truth diffusion and jump functions $\sigma(X,t) = 0.1\sqrt{|X|}, \beta(X,t) = 0.1\sqrt{|X|}$ are small.

If no prior information on equation (59) is given, the average errors for the reconstructed drift, diffusion, and jump functions are 1.412, 0.790, and 0.347, respectively. This high error might arise from training set trajectories that are too noisy or sparsely distributed. However, if the drift function is given, the diffusion and jump functions can be much more accurately reconstructed, leading to relative errors below 0.2 for all three forms of $\sigma(X_t,t)$ and $\beta(X_t,t)$ used to define the ground truth (see figures 2(d)–(k)). On the other hand, providing the diffusion or jump function does not improve the accuracy of the reconstruction of the other unknown functions in equation (59). The average errors of the reconstructed diffusion and jump functions, when different prior information is given, are listed in table 2.

In appendix I, we carry out an additional numerical experiment by varying the number of trajectories in the training set. The errors in the reconstructed drift, diffusion, and jump function decrease when the number of trajectories for training increases without any prior information. This indicates that our temporally decoupled squared W_2 method has the potential to accurately reconstruct equation (59) even without prior information provided there are a sufficient number of training trajectories. On the other hand, if the drift function is given as prior information, the errors of the reconstructed diffusion and jump functions are around 0.2 even when only 100 trajectories are used. Therefore, information on the drift function can significantly boost the performance of our temporally decoupled squared W_2 method, allowing accurate reconstruction of equation (59) even when the number of observed trajectories is limited.

In real physical systems, the drift function can often be obtained by measurements over a macroscopic ensemble of trajectories, such as mass-action kinetics if the $X(t)$ in equation (1) denotes some physical quantity, e.g. the number density of molecules [32, 33]. Thus, after independently measuring the drift function and inputting it as a prior knowledge, our temporally decoupled squared W_2-distance method can be used to reconstruct the diffusion and jump functions efficiently.

We carry out an extra numerical experiment reconstructing equation (59) by varying σ_0, β_0 in equations (60)–(62). With the drift function provided, our temporally decoupled squared W_2-distance method can accurately reconstruct the diffusion and the jump functions for different values of σ_0 and β_0 in equations (60)–(62). The results are shown in appendix J.

In our last example, we test whether our temporally decoupled squared W_2-distance can accurately reconstruct a 2D jump-diffusion process with correlated Brownian-type and compensated-Poisson-type noise across the two stochastic variables.

Example 4.3. We reconstruct the following 2D jump-diffusion process, which is obtained by superimposing a 2D compensated Poisson process $\tilde{\boldsymbol{N}}_t := (\tilde{N}_1(t), \tilde{N}_2(t))$ onto the pure diffusion process that describes the dynamics of a synthetic data set characterizing gene regulatory dynamics in biophysics [4, 34]:

$$\mathrm{d}\boldsymbol{X}(t) = -\boldsymbol{g}(\boldsymbol{X}(t))\,\mathrm{d}t + \boldsymbol{\sigma}(\boldsymbol{X}(t))\,\mathrm{d}\boldsymbol{W}_t + \boldsymbol{\beta}(\boldsymbol{X}(t))\,\mathrm{d}\tilde{\boldsymbol{N}}_t,\ \boldsymbol{X}(t=0) = \boldsymbol{X}_0,\ t \in [0,T]\,. \tag{63}$$

$\tilde{N}_1(t)$ and $\tilde{N}_2(t)$ are independent and both have intensity t. Here, $\boldsymbol{X}(t) = (X_1(t), X_2(t)) \in \mathbb{R}^2$, $\boldsymbol{g}(\boldsymbol{X}) : \mathbb{R}^2 \to \mathbb{R}^2$ is the drift function, and $\boldsymbol{\sigma}, \boldsymbol{\beta} : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ are the diffusion and jump functions, respectively. The drift function $\boldsymbol{g}$ is given by

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{X}) = &\left(\frac{1}{\sigma_1}\frac{N_1}{N_1+N_2}(X_1-\mu_{11})+\frac{1}{\sigma_2}\frac{N_2}{N_1+N_2}(X_1-\mu_{21}),\right.\\ &\left.\frac{1}{\sigma_1}\frac{N_1}{N_1+N_2}(X_2-\mu_{21})+\frac{1}{\sigma_2}\frac{N_2}{N_1+N_2}(X_2-\mu_{22})\right)^T,\\ N_1 := N_1(\boldsymbol{X}) = &\frac{1}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{(X_1-\mu_{11})^2}{2\sigma_1^2}-\frac{(X_2-\mu_{12})^2}{2\sigma_1^2}\right),\\ N_2 := N_2(\boldsymbol{X}) = &\frac{1}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{(X_1-\mu_{21})^2}{2\sigma_2^2}-\frac{(X_2-\mu_{22})^2}{2\sigma_2^2}\right).\end{aligned} \tag{64}$$

The parameters are set as $\sigma_1 = 1, \sigma_2 = 0.95, \mu_{11} = 1.6, \mu_{12} = 1.2, \mu_{21} = 1.8, \mu_{22} = 1.0$. We set $T = 10.2$ and an initial condition $\boldsymbol{X}_0 = (1.7, 1.1)$. We take the correlated diffusivity as

$$\boldsymbol{\sigma} = \begin{bmatrix}\sigma_0\sqrt{|X_1|} & c_1\sigma_0\sqrt{|X_2|}\\ c_1\sigma_0\sqrt{|X_1|} & \sigma_0\sqrt{|X_2|}\end{bmatrix}, \tag{65}$$

and the jump function of the compensated Poisson process as

$$\boldsymbol{\beta} = \begin{bmatrix}\beta_0 & c_2\beta_0\\ c_2\beta_0 & \beta_0\end{bmatrix}. \tag{66}$$

Here, c_1 and c_2 determine the correlations of Brownian noise and compensated Poisson process across the two dimensions, respectively. Specifically, when $c_1 = 0$ (or $c_2 = 0$), the Brownian (or compensated Poisson) noise in each variable is independent of the other; when $c_1 = 1$ (or $c_2 = 1$), the Brownian-type (or compensated-Poisson-type) noise across the two dimensions are linearly dependent; when $c_1 = -1$ (or $c_2 = -1$), the Brownian-type (or compensated-Poisson-type) noise across the two dimensions are perfectly negatively correlated.

From example 4.2, imposing a prior on the drift function can greatly improve the accuracy of the reconstructed diffusion and jump functions. Thus, we input $\boldsymbol{g}(\boldsymbol{X})$ defined in equation (64) as prior information. Since the jump-diffusion process described by equation (63) is two-dimensional, we use the following error metric to measure the errors in the diffusion and jump functions:

$$\text{diffusion error} = \frac{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\|\boldsymbol{\sigma}\boldsymbol{\sigma}^T(x_j(t_i),t_i)-\hat{\boldsymbol{\sigma}}\hat{\boldsymbol{\sigma}}^T(x_j(t_i),t_i)\|_F^2}{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\|\hat{\boldsymbol{\sigma}}\hat{\boldsymbol{\sigma}}^T(x_j(t_i),t_i)\|_F{}^2}; \tag{67}$$

$$\text{jump error} = \frac{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\|\boldsymbol{\beta}\boldsymbol{\beta}^T(x_j(t_i),t_i)-\hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T(x_j(t_i),t_i)\|_F^2}{\sum_{i=0}^{N}\sum_{j=1}^{M_s}\|\hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T(x_j(t_i),t_i)\|_F{}^2}. \tag{68}$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. We set $\sigma_0 = 0.1, \beta_0 = 0.1$ in equations (65) and (66). Different values of c_1, c_2 are used to tune the correlations to explore how they affect the reconstruction of the jump-diffusion process.

Figures 3(a)–(c) show that solutions generated by our reconstructed jump-diffusion process with the temporally decoupled squared W_2 loss function match well with solutions generated by the 2D jump-diffusion process equation (63) ($c_1 = c_2 = -0.5$ in equations (65) and (66)). Figures 3(d)–(e) indicate that when the drift function $\boldsymbol{g}(\boldsymbol{X}(t))$ is given, our temporally decoupled squared W_2 method can accurately reconstruct the diffusion and jump functions for most combinations of c_1, c_2 The average errors in the diffusion and jump functions averaged over all combinations of c_1, c_2 are 0.197 and 0.210, respectively. Also, the final distribution of the reconstructed $\hat{\boldsymbol{X}}(t)$ aligns well with the ground truth $\boldsymbol{X}(t)$.

In appendix K, we implement our reconstruction method by using different numbers of hidden layers and different numbers of neurons in each layer for the neural-network-parameterized approximation to the diffusion and jump functions $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$. We find that with the drift function given as prior information, increasing the number of neurons per layer can improve the accuracy of the reconstructed diffusion and the jump function of the 2D jump-diffusion process equation (63). Increasing the number of hidden layers also leads to a more accurate reconstruction of the diffusion and jump function when the number of hidden layers is smaller than three; however, after three hidden layers, increasing their number leads to less accuracy of the reconstructed $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$. Setting the number of hidden layers to three and the number of neurons per layer to about 400 leads to excellent reconstruction of $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$. However, larger numbers of hidden layers or neurons per hidden layer demand more memory usage and lead to longer runtimes. We also found that implementing Dropout layers [35, 36] did not improve the accuracy of reconstructing the diffusion and jump functions. This could be

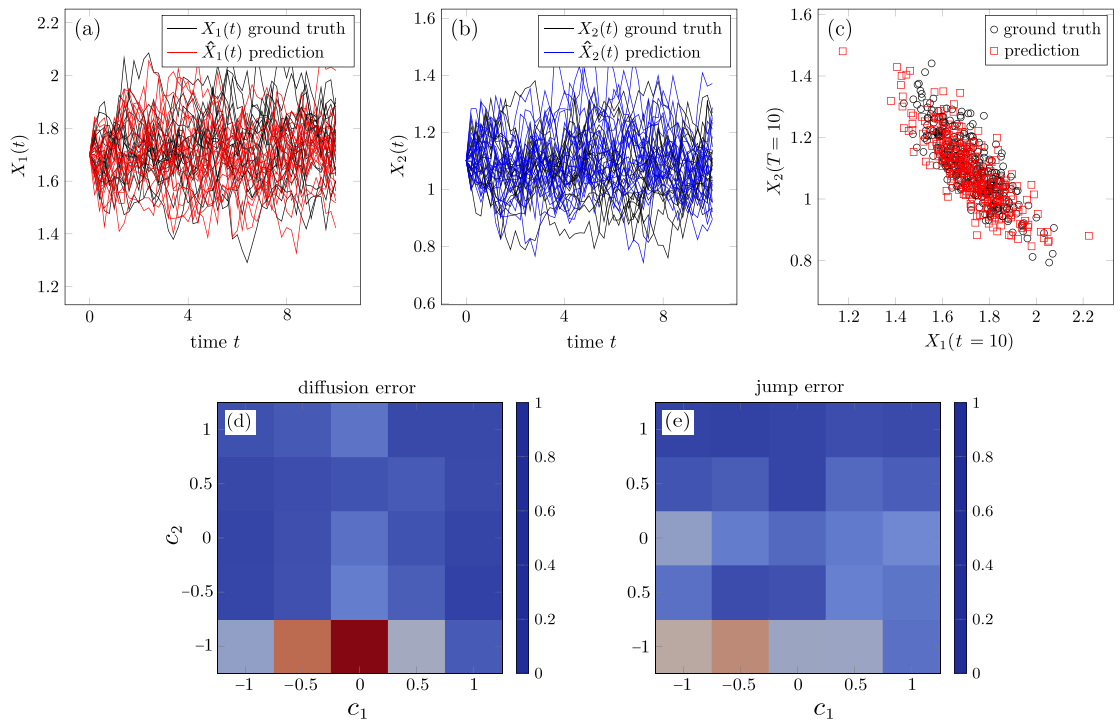

Figure 3. (a) and (b) Solutions generated by the reconstructed jump-diffusion process using our temporally decoupled squared W_2 method versus solutions generated by the ground truth equation (63). (c) The reconstructed $\hat{\boldsymbol{X}}(t=10)$ versus the ground truth $\boldsymbol{X}(t=10)$. In (a)–(c), $c_1 = c_2 = -0.5$. (d) The error (equation (67)) between the ground truth diffusion function $\boldsymbol{\sigma}$ and the reconstructed diffusion function $\hat{\boldsymbol{\sigma}}$. (e) The error (equation (68)) between the ground truth jump function $\boldsymbol{\beta}$ and the reconstructed diffusion function $\hat{\boldsymbol{\beta}}$. In (d) and (e), the errors are averaged over 5 independent experiments.

because the Dropout technique, by randomly ignoring neurons in hidden layers during training, introduces stochasticity to the reconstructed diffusion and jump functions during training on top of the intrinsic noise in the jump-diffusion process, interfering with the accurate reconstruction of the inherently deterministic diffusion and jump functions. The network architecture required for optimal reconstruction of diffusion and jump functions warrants further investigation.

5. Summary & conclusions

In this paper, we proposed and showed how to use a temporally decoupled squared W_2-distance $\tilde{W}_2^2(\mu, \hat{\mu})$ defined in equation (22) in the reconstruction of jump-diffusion processes. Minimization of this Wasserstein-distance-based loss function leads to small errors in the drift, diffusion, and jump functions $\boldsymbol{f} - \hat{\boldsymbol{f}}$, $\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}$, and $\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}$, when approximating a jump-diffusion process (equation (1)) by another jump-diffusion process (equation (5)). Moreover, the temporally decoupled squared W_2-distance can be efficiently evaluated using finite-sample finite-time-point observations.

Through several numerical experiments, we showed that minimizing our proposed temporally decoupled squared W_2-distance loss performs much better than other commonly used loss functions and methods for jump-diffusion process reconstruction using parameterized neural networks. Furthermore, we showed that if we impose prior knowledge on the drift function, the diffusion and jump functions can be more accurately reconstructed.

Our approach can potentially be extended and applied to other reconstruction problems in physics, such as those involving higher-dimensional dynamical or oscillatory and chaotic systems [37, 38] to investigate whether observed 'chaotic' dynamics result from a chaotic ODE system or the intrinsic stochasticity of a jump-diffusion process instead. The Wasserstein distance can also be adapted for reconstructing other stochastic processes such as Lévy walks involving compound Poisson process [39, 40]. Reconstructing such processes could require inferring the intensity of the Poisson process, which is nontrivial and would require consideration of differentiation w.r.t. 'discrete randomness' [41]. The reconstruction of jump-diffusion

processes from trajectories of particle motion can be used to detect anomalous superdiffusive or subdiffusive dynamics, an important paradigm in fields as diverse as astrophysics [42] and materials science [43].

Finally, partial prior knowledge may improve the reconstruction of the underlying jump-diffusion process (equation (1)) and even pure-diffusion processes [44] using physics-informed learning method. For example, if the symmetry of the drift function in equation (1) is known, the symmetry can be encoded in the neural network architecture to improve the reconstruction accuracy as in physics-informed neural networks. Alternatively, if the drift function is known to be a specific function $f(x;\alpha)$ up to an unknown parameter α, then we can directly incorporate $f(x;\alpha)$ as the drift function into equation (5) rather than a neural network approximation. The true α can be directly learned by backpropagation and gradient descent. In the context of physical models, it may be fruitful to explore how partial prior knowledge facilitates the reconstruction of jump-diffusion processes.

Data availability statement

An efficient Wasserstein-distance approach for reconstructing jump-diffusion processes using parameterized neural networks. All codes are publicly available at https://github.com/mtxia99/jump_diffusion_process_reconstruction. No new data were created or analysed in this study.

Conflict of interest

The authors declare that they have no conflicts of interest to report regarding the present study.

Appendix A. Proof to theorem 2.1

Here, we provide proof for theorem 2.1. Our strategy is similar to that used in the proof of the stochastic Gronwall lemma (theorem 2.2 in [6]). First, we apply the Ito's lemma to

$$\begin{aligned}\left|X_i(t)-\tilde{X}_i(t)\right|^2 &= 2\int_0^t \left(X_i(s^-)-\tilde{X}_i(s^-)\right)\left(f_i\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{f}_i\left(\tilde{\boldsymbol{X}}(s^-),s^-\right)\right)\mathrm{d}s\\ &\quad+2\int_0^t\sum_{j=1}^m \left(X_i(s^-)-\tilde{X}_i(s^-)\right)\left(\sigma_{i,j}\left(\boldsymbol{X}(s^-),s^-\right)-\hat{\sigma}_{i,j}\left(\tilde{\boldsymbol{X}}_i(s^-),s^-\right)\right)\mathrm{d}B_{j,s}\\ &\quad+2\int_0^t\int_U \left(X_i(s^-)-\tilde{X}_i(s^-)\right)\left(\beta_i\left(\boldsymbol{X}(s^-),\xi,s^-\right)-\hat{\beta}_i\left(\hat{\boldsymbol{X}}(s^-),\xi(s^-)\right)\right)\mathrm{d}\tilde{N}\left(\mathrm{d}s,\nu(\mathrm{d}\xi)\right)\\ &\quad+\int_0^t\sum_{j=1}^m\left(\sigma_{i,j}\left(X_i(s^-),s^-\right)-\hat{\sigma}_{i,j}\left(\tilde{X}_i(s^-),s^-\right)\right)^2\mathrm{d}s\\ &\quad+\int_0^t\int_U\left(\beta_i\left(X_i(s^-),\xi,s^-\right)-\hat{\beta}_{i,j}\left(\tilde{X}_i(s^-),\xi,s^-\right)\right)^2\nu(\mathrm{d}\xi)\mathrm{d}s.\end{aligned} \tag{A.1}$$

Note that

$$\begin{aligned}f_i\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{f}_i\left(\tilde{\boldsymbol{X}}\left(s^-\right),s^-\right) &= \left(f_i\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{f}_i\left(\boldsymbol{X}\left(s^-\right),s^-\right)\right)\\ &\quad+\left(\hat{f}_i\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{f}_i\left(\tilde{\boldsymbol{X}}\left(s^-\right),s^-\right)\right),\\ \sigma_{i,j}\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{\sigma}_{i,j}\left(\tilde{\boldsymbol{X}}\left(s^-\right),s^-\right) &= \left(\sigma_{i,j}\left(\boldsymbol{X}\left(s^-\right),s^-\right)-\hat{\sigma}_{i,j}\left(\boldsymbol{X}\left(s^-\right),s^-\right)\right)\\ &\quad+\left(\hat{\sigma}_{i,j}(\boldsymbol{X}(s^-),s^-)-\hat{\sigma}_{i,j}(\tilde{\boldsymbol{X}}(s^-),s^-)\right),\\ \beta_i(\boldsymbol{X}(s^-),\xi,s^-)-\hat{\beta}_i(\tilde{\boldsymbol{X}}(s^-),\xi,s^-) &= \left(\beta_i(\boldsymbol{X}(s^-),\xi,s^-)-\hat{\beta}_i(\boldsymbol{X}(s^-),\xi,s^-)\right)\\ &\quad+\left(\hat{\beta}_i(\boldsymbol{X}(s^-),\xi,s^-)-\hat{\beta}_i(\tilde{\boldsymbol{X}}(s^-),\xi,s^-)\right).\end{aligned} \tag{A.2}$$

Using the Lipschitz conditions on the drift, diffusion, and jump functions $\hat{\boldsymbol{f}}, \hat{\boldsymbol{\sigma}}$, and $\hat{\boldsymbol{\beta}}$ in assumption 2.1 and the Cauchy inequality, from equations (A.1) and (A.2), we find

$$\begin{aligned}
\left|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\right|_2^2 &= \sum_{i=1}^{d}\left(X_i(t)-\hat{X}_i(t)\right)^2 \\
&\leqslant \left(2\bar{f}+2\overline{\sigma}m+2\nu(U)\overline{\beta}+1+m+\nu(U)\right)\sum_{i=1}^{d}\int_0^t\left(X_i(s^-)-\tilde{X}_i(s^-)\right)^2\mathrm{d}s \\
&\quad+\sum_{i=1}^{d}\int_0^t\left|f_i(\boldsymbol{X}(s^-),s^-)-\hat{f}_i\boldsymbol{X}(s^-),s^-)\right|^2\mathrm{d}s \\
&\quad+2\sum_{i=1}^{d}\int_0^t\sum_{j=1}^{m}\left|\sigma_{i,j}(\boldsymbol{X}(s^-),s^-)-\hat{\sigma}_{i,j}(\boldsymbol{X}(s^-),s^-)\right|^2\mathrm{d}s \\
&\quad+2\sum_{i=1}^{d}\int_0^t\int_U\left|\beta_i(\boldsymbol{X}(s^-),\xi,s^-)-\hat{\beta}_i(\boldsymbol{X}(s^-),\xi,s^-)\right|^2\nu(\mathrm{d}\xi)\mathrm{d}s \\
&\quad+2\sum_{i=1}^{d}\int_0^t\sum_{j=1}^{m}\left(X_i(s^-)-\tilde{X}_i(s^-)\right)\left(\sigma_{i,j}(\boldsymbol{X}(s^-),s^-)-\hat{\sigma}_{i,j}(\tilde{\boldsymbol{X}}_i(s^-),s^-)\right)\mathrm{d}B_{j,s} \\
&\quad+2\sum_{i=1}^{d}\int_0^t\int_U\left(X_i(s^-)-\tilde{X}_i(s^-)\right)\left(\beta_i(\boldsymbol{X}(s^-),\xi,s^-)-\hat{\beta}_i(\tilde{\boldsymbol{X}}(s^-),\xi,s^-)\right)\mathrm{d}\tilde{N}\left(\mathrm{d}s,\nu(\mathrm{d}\xi)\right).
\end{aligned} \tag{A.3}$$

From assumption 2.1 and the conditions in theorem 2.1, the second, third, and fourth terms on the RHS of equation (A.3) are adapted and non-decreasing w.r.t. t; the fifth and sixth terms on the RHS of equation (A.3) are martingales. Thus, by taking the expectation of both sides of equation (A.3), we find

$$\mathbb{E}\left[\left|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\right|_2^2\right]\leqslant\left(2\bar{f}+1+(2\overline{\sigma}+1)m+\left(2\overline{\beta}+1\right)\nu(U)\right)\int_0^t\mathbb{E}\left[\left|\boldsymbol{X}(s)-\tilde{\boldsymbol{X}}(s)\right|_2^2\right]\mathrm{d}s+\mathbb{E}\left[H(t)\right], \tag{A.4}$$

where $H(t)$ is defined in equation (14). Applying Gronwall's lemma to $u(t):=\mathbb{E}\left[\left|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\right|_2^2\right]$ and noticing that $\mathbb{E}[H(t)]$ is non-decreasing w.r.t. t, we conclude that

$$u(t)\leqslant\mathbb{E}\left[\left|\boldsymbol{X}(t)-\tilde{\boldsymbol{X}}(t)\right|^2\middle|\boldsymbol{X}(0)\right]\leqslant\exp\left(\left(2\bar{f}+1+(2\overline{\sigma}+1)m+\left(2\overline{\beta}+1\right)\nu(E)\right)T\right)\cdot\mathbb{E}\left[H(T)\,|\boldsymbol{X}(0)\right], \tag{A.5}$$

which proves equation (13).

Appendix B. Proof to theorem 2.2

Here, we shall provide proof of theorem 2.2, which generalizes theorem 2 in [25] for pure diffusion processes. Denote

$$\Omega_N:=\{\boldsymbol{Y}(t)|\boldsymbol{Y}(t)=\boldsymbol{Y}(t_i)\ \ t\in[t_i,t_{i+1}),i<N-1;\ \boldsymbol{Y}(t)=\boldsymbol{Y}(t_i),\ \ t\in[t_i,t_{i+1}]\} \tag{B.1}$$

to be the space of piecewise functions. Clearly, it is a subspace of $L^2([0,T];\mathbb{R}^d)$. Also, the embedding map from Ω_N to $L^2([0,T];\mathbb{R}^d)$ preserves the $\|\cdot\|$ norm, which enables us to define the measures on $\mathcal{B}(L^2([0,T];\mathbb{R}^d))$ induced by the measures $\mu_N,\hat{\mu}_N$. For simplicity, we shall still denote those induced measures by $\mu_N,\hat{\mu}_N$.

Suppose $\boldsymbol{X}(t),\hat{\boldsymbol{X}}(t)$ are generated by two jump-diffusion processes defined by equations (1) and (5). The inequality equation (19) is a direct result of the triangular inequality for the Wasserstein distance [45] because $\boldsymbol{X},\boldsymbol{X}_N,\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}_N\in L^2([0,T];\mathbb{R}^d)$.

Next, we prove equation (21). Because $\boldsymbol{X}_N(t)=I_N\boldsymbol{X}(t)$ (defined in equation (18)), we choose a specific *coupling measure*, i.e. the coupled distribution, π of μ,μ_N that is essentially the 'original' probability distribution. To be more specific, for an abstract probability space $(\Omega,\mathcal{A},p)$ associated with $\boldsymbol{X}$, μ and μ_N can be characterized by the *pushforward* of p via $\boldsymbol{X}$ and $\boldsymbol{X}_N$ respectively, i.e. $\mu=\boldsymbol{X}_*p$, defined by $\forall A\in\mathcal{B}\left(\tilde{\Omega}_N\right)$, elements in the Borel σ-algebra of $\tilde{\Omega}_N$,

$$\mu(A)=\boldsymbol{X}_*p(A):=p\left(\boldsymbol{X}^{-1}(A)\right), \tag{B.2}$$

where $\boldsymbol{X}$ is interpreted as a measurable map from Ω to $\tilde{\Omega}_N$, and $\boldsymbol{X}^{-1}(A)$ is the preimage of A under $\boldsymbol{X}$. Then, the coupling π is defined by

$$\pi = (\boldsymbol{X}, \boldsymbol{X}_N)_* p, \tag{B.3}$$

where $(\boldsymbol{X}, \boldsymbol{X}_N)$ are interpreted as a measurable map from Ω to $\tilde{\Omega}_N \times \tilde{\Omega}_N$. One can readily verify that the marginal distributions of π are μ and μ_N respectively. Therefore, the squared $W_2^2(\mu, \mu_N)$ can be bounded by`

$$W_2^2(\mu, \mu_N) \leqslant \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left|\boldsymbol{X}(t) - \boldsymbol{X}_N(t)\right|_2^2\right] \mathrm{d}t = \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \sum_{\ell=1}^{d} \mathbb{E}\left[\left(X_\ell(t) - X_{N,\ell}(t)\right)^2\right] \mathrm{d}t. \tag{B.4}$$

For each $\ell = 1, \ldots, d$, by using the Itô's isometry and the orthogonality condition of the compensated Poisson process $\tilde{N}$ (in assumption 2.1), we have

$$\begin{aligned}
\sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left(X_\ell(t) - X_{N,\ell}(t)\right)^2\right] \mathrm{d}t \leqslant & \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left(\int_{t_i}^{t} f_\ell\left(\boldsymbol{X}\left(r^-\right), r^-\right) \mathrm{d}r\right)^2\right] \mathrm{d}t \\
& + \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left(\int_{t_i}^{t} \sum_{j=1}^{m} \sigma_{\ell,j}\left(\hat{X}(r^-), r^-\right) \mathrm{d}B_{j,r}\right)^2\right] \mathrm{d}t \\
& + \sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left(\int_{t_i}^{t} \int_{U} \beta_\ell\left(\boldsymbol{X}\left(r^-\right), \xi, r^-\right) \tilde{N}(\mathrm{d}r, \nu(\mathrm{d}\xi))\right)^2\right] \mathrm{d}t \\
\leqslant & \sum_{i=1}^{N} \left(\Delta t_{i-1}\right)^2 \mathbb{E}\left[\int_{t_{i-1}}^{t_i} f_\ell^2 \mathrm{d}t\right] + \sum_{i=1}^{N} \Delta t_{i-1} \sum_{j=1}^{m} \mathbb{E}\left[\int_{t_{i-1}}^{t_i} \sigma_{\ell,j}^2 \mathrm{d}t\right] \\
& + \sum_{i=1}^{N} \Delta t_{i-1} \mathbb{E}\left[\int_{t_{i-1}}^{t_i} \int_{U} \beta_\ell^2 \nu(\mathrm{d}\xi)\, \mathrm{d}t\right],
\end{aligned} \tag{B.5}$$

where $\Delta t_{i-1} := t_i - t_{i-1}$. Summing over ℓ, we have

$$\sqrt{\sum_{i=1}^{N} \int_{t_{i-1}}^{t_i} \mathbb{E}\left[\left|\boldsymbol{X}(t) - \boldsymbol{X}_N(t)\right|_2^2\right] \mathrm{d}t} \leqslant \sqrt{F \Delta t^2 + \Sigma \Delta t + B \Delta t}, \tag{B.6}$$

where $\Delta t := \max_{0 \leqslant i \leqslant N-1}(t_{i+1} - t_i)$. Similarly, we can show that

$$W_2(\hat{\mu}, \hat{\mu}_N) \leqslant \sqrt{\hat{F} \Delta t^2 + \hat{\Sigma} \Delta t + \hat{B} \Delta t}. \tag{B.7}$$

Plugging equations (B.6) and (B.7) into equation (19), we have proved equation (21).

Appendix C. Proof to theorem 3.1

Here, we provide proof to theorem 3.1. The proof builds upon and generalizes the proof of theorem 3 in [25] for pure diffusion processes to jump-diffusion processes. First, notice that

$$\mathbb{E}\left[\left|\boldsymbol{X}(t) - \hat{\boldsymbol{X}}(t)\right|_2^2\right] \leqslant 2\left(FT + \hat{F}T + \Sigma + \hat{\Sigma} + B + \hat{B}\right) < \infty, \ \forall t \in [0, T] \tag{C.1}$$

where $F, \hat{F}, \Sigma, \hat{\Sigma}, B, \hat{B}$ are defined in equation (20). By applying theorem 2.2, for any $t_i, i = 1, 2, \ldots, N$, denoting $\Delta t_i := t_i - t_{i-1}$, we have

$$\begin{aligned}
\inf_{\pi_i} \sqrt{\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)\right|_2^2\right] \Delta t_i} - \sqrt{F_i \left(\Delta t_i\right)^2 + \Sigma_i \Delta t_i + B_i \Delta t_i} - \sqrt{\hat{F}_i \left(\Delta t_i\right)^2 + \hat{\Sigma}_i \Delta t_i + \hat{B}_i \Delta t_i} & \\
\leqslant W_2\left(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i\right) & \\
\leqslant \inf_{\pi_i} \sqrt{\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)\right|_2^2\right] \Delta t_i} + \sqrt{F_i \left(\Delta t_i\right)^2 + \Sigma_i \Delta t_i + B_i \Delta t_i} & \\
+ \sqrt{\hat{F}_i \left(\Delta t_i\right)^2 + \hat{\Sigma}_i \Delta t_i + \hat{B}_i \Delta t_i}, &
\end{aligned} \tag{C.2}$$

where $\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i$ are the distributions for $\boldsymbol{X}(t), t \in [t_i, t_{i+1})$ and $\hat{\boldsymbol{X}}(t), t \in [t_i, t_{i+1})$, respectively. Additionally, from equation (30), we have

$$\begin{aligned}
&\sum_{i=0}^{N-1} F_i = F < \infty, \ \sum_{i=0}^{N-1} \Sigma_i = \Sigma < \infty, \ \sum_{i=0}^{N-1} B_i = B < \infty \\
&\sum_{i=0}^{N-1} \hat{F}_i = \hat{F} < \infty, \ \sum_{i=0}^{N-1} \hat{\Sigma}_i = \hat{\Sigma} < \infty, \ \sum_{i=0}^{N-1} \hat{B}_i = \hat{B} < \infty.
\end{aligned} \tag{C.3}$$

From the inequality (C.2), we have

$$\begin{aligned}
W_2^2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \leqslant & \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i \\
& + 2 \inf_{\pi_i} \sqrt{\mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i} \left[\sqrt{F_i \Delta t_i + \Sigma_i + B_i} + \sqrt{\hat{F} \Delta t_i + \hat{\Sigma}_i + \hat{B}_i}\right] \\
& + 2\Delta t_i \left(F_i \Delta t_i + \Sigma_i + B_i + \hat{F}_i \Delta t_i + \hat{\Sigma}_i + \hat{B}_i\right) \\
W_2^2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \geqslant & \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i \\
& - 2W_2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \Delta t_i \left[\sqrt{F_i \Delta t_i + \Sigma_i + B_i} + \sqrt{\hat{F} \Delta t_i + \hat{\Sigma}_i + \hat{B}_i}\right] \\
& - 2\Delta t_i \left(F_i \Delta t_i + \Sigma_i + B_i + \hat{F}_i \Delta t_i + \hat{\Sigma}_i + \hat{B}_i\right).
\end{aligned} \tag{C.4}$$

Specifically, from the assumption given in equations (C.1) and (C.2), we conclude that

$$W_2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \leqslant \sqrt{\Delta t_i} \left(M + \sqrt{F \Delta t_i + \Sigma_i + B_i} + \sqrt{\hat{F} \Delta t_i + \hat{\Sigma}_i + \hat{B}_i}\right) := \tilde{M} \sqrt{\Delta t_i}, \ \tilde{M} < \infty. \tag{C.5}$$

Summing over $i = 1, \ldots, N-1$ for both inequalities in equation (C.4) and noting that $\Delta t = \max_i |t_{i+1} - t_i|$, we conclude

$$\begin{aligned}
\sum_{i=0}^{N-1} W_2^2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \leqslant & \sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i + 2\Delta t \left(F \Delta t + \Sigma + \hat{F} \Delta t + \hat{\Sigma} + B + \hat{B}\right) \\
& + 2M \sum_{i=1}^{N-1} \Delta t_i \left(\sqrt{F_i \Delta t_i + \Sigma_i + B_i} + \sqrt{\hat{F}_i \Delta t_i + \hat{\Sigma}_i + \hat{B}_i}\right), \\
\leqslant & \sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i + 2\Delta t \left(F \Delta t + \Sigma + \hat{F} \Delta t + \hat{\Sigma} + B + \hat{B}\right) \\
& + M \sqrt{\Delta t} \left(\left(F + \hat{F}\right) \Delta t + \Sigma + \hat{\Sigma} + B + \hat{B} + 2T\right)
\end{aligned} \tag{C.6}$$

and

$$\begin{aligned}
\sum_{i=0}^{N-1} W_2^2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \geqslant & \sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i \\
& - 2\tilde{M} \sum_{i=0}^{N-1} \Delta t_i \left(\sqrt{F_i \Delta t_i + \Sigma_i + B_i} + \sqrt{\hat{F}_i \Delta t_i + \hat{\Sigma}_i + \hat{B}_i}\right) \\
& - 2\Delta t \left(F \Delta t + \Sigma + B + \hat{F} \Delta t + \hat{\Sigma} + \hat{B}\right), \\
\geqslant & \sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i - 2\Delta t \left(F \Delta t + \Sigma + B + \hat{F} \Delta t + \hat{\Sigma} + \hat{B}\right) \\
& - \tilde{M} \sqrt{\Delta t} \left(\left(F + \hat{F}\right) \Delta t + \Sigma + \hat{\Sigma} + B + \hat{B} + 2T\right).
\end{aligned} \tag{C.7}$$

Equations (C.6) and (C.7) indicate that as $N \to \infty$,

$$\sum_{i=0}^{N-1} \inf_{\pi_i} \mathbb{E}_{\pi_i}\left[|\boldsymbol{X}(t_i) - \hat{\boldsymbol{X}}(t_i)|_2^2\right] \Delta t_i - \sum_{i=0}^{N-1} W_2^2(\boldsymbol{\mu}_i, \hat{\boldsymbol{\mu}}_i) \to 0, \tag{C.8}$$

which proves equation (29).

Now, suppose $0=t_0^1<t_1^1<\ldots<t_{N_1}^1=T$; $0=t_0^2<t_1^2<\ldots<t_{N_2}^2=T$ to be 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 $\mu_i^s(t_i^1)$ and $\hat{\mu}_i^s(t_i^1)$ to be the probability distribution of $\boldsymbol{X}(t_i^s)$ and $\hat{\boldsymbol{X}}(t_i^s)$, $s=1,2,3$, respectively. We now prove that

$$\left|\sum_{i=0}^{N_1-1}W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_{i+1}^1-t_i^1\right)-\sum_{i=0}^{N_3-1}W_2^2\left(\mu\left(t_i^3\right),\hat{\mu}\left(t_i^3\right)\right)\left(t_{i+1}^3-t_i^3\right)\right|\to 0, \tag{C.9}$$

as $\Delta t\to 0$.

First, suppose in the interval (t_i^1,t_{i+1}^1), we have $t_i^1=t_\ell^3<t_{\ell+1}^3<\ldots<t_{\ell+s}^3=t_{i+1}^1, s\geqslant 1$, then 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

$$\begin{aligned}&\left|W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_{i+1}^1-t_i^1\right)-\sum_{k=\ell}^{\ell+s-1}W_2^2\left(\mu\left(t_k^3\right),\hat{\mu}\left(t_i^3\right)\right)\left(t_{k+1}^3-t_k^3\right)\right|\\&\leqslant\sum_{k=\ell+1}^{\ell+s-1}\left(W_2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)+W_2\left(\hat{\mu}\left(t_i^3\right),\hat{\mu}\left(t_k^3\right)\right)\right)\\&\quad\times\left(W_2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)-W_2\left(\mu\left(t_k^3\right),\hat{\mu}\left(t_k^3\right)\right)\right)\left(t_{k+1}^3-t_k^3\right).\end{aligned} \tag{C.10}$$

On the other hand, because we can take a specific coupling π^* to be the joint distribution of $(\boldsymbol{X}(t_i^1),\boldsymbol{X}(t_k^3))$,

$$\begin{aligned}W_2\left(\mu\left(t_i^1\right),\mu\left(t_k^3\right)\right)&\leqslant\sqrt{\mathbb{E}\left[\left|\boldsymbol{X}\left(t_k^3\right)-\boldsymbol{X}\left(t_i^1\right)\right|_2^2\right]}\\&\leqslant\mathbb{E}\left[\int_{t_i^1}^{t_{i+1}^1}\sum_{i=1}^{d}f_i^2\left(\boldsymbol{X}\left(t^-\right),t^-\right)\mathrm{d}t+\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^{d}\sum_{j=1}^{m}\sigma_{\ell,j}^2\left(\boldsymbol{X}\left(t^-\right),t^-\right)\mathrm{d}t\right.\\&\quad\left.+\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^{d}\int_U\beta_\ell^2\left(\boldsymbol{X}\left(t^-\right),\xi,t^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}t\right]^{\frac{1}{2}}.\end{aligned} \tag{C.11}$$

Similarly, we have

$$\begin{aligned}W_2\left(\hat{\mu}\left(t_i^1\right),\hat{\mu}\left(t_k^3\right)\right)\leqslant\mathbb{E}&\left[\int_{t_i}^{t_{i+1}}\sum_{\ell=1}^{d}\hat{f}_\ell^2\left(\boldsymbol{X}\left(t^-\right),t^-\right)\mathrm{d}t+\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^{d}\sum_{j=1}^{m}\hat{\sigma}_{\ell,j}^2\left(\boldsymbol{X}\left(t^-\right),t^-\right)\mathrm{d}t\right.\\&\left.+\int_{t_i^1}^{t_{i+1}^1}\sum_{\ell=1}^{d}\int_U\hat{\beta}_\ell^2\left(\hat{\boldsymbol{X}}\left(t^-\right),\xi,t^-\right)\nu\left(\mathrm{d}\xi\right)\mathrm{d}t\right]^{\frac{1}{2}}.\end{aligned} \tag{C.12}$$

Using the triangular inequality of the Wasserstein distance as well as the Cauchy inequality, we have

$$\begin{aligned}\left|W_2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)-W_2\left(\mu\left(t_k^3\right),\hat{\mu}\left(t_k^3\right)\right)\right|&\leqslant\left|W_2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)-W_2\left(\mu\left(t_k^3\right),\hat{\mu}\left(t_i^1\right)\right)\right|\\&\quad+\left|W_2\left(\mu\left(t_i^3\right),\hat{\mu}\left(t_i^1\right)\right)-W_2\left(\mu\left(t_k^3\right),\hat{\mu}\left(t_k^3\right)\right)\right|\\&\leqslant W_2\left(\mu\left(t_i^1\right),\mu\left(t_k^3\right)\right)+W_2\left(\hat{\mu}\left(t_i^1\right),\hat{\mu}\left(t_k^3\right)\right).\end{aligned} \tag{C.13}$$

Substituting equations (C.11), (C.12), (C.1) and (C.13) into equation (C.10), we conclude that

$$\begin{aligned}&\left|W_2^2\left(\mu(t_i^1),\hat{\mu}(t_i^1)\right)\left(t_{i+1}^1-t_i^1\right)-\sum_{k=\ell}^{\ell+s-1}W_2^2\left(\left(\mu(t_k^3),\hat{\mu}(t_k^3)\right)\left(t_{k+1}^3-t_k^3\right)\right)\right|\\&\leqslant 2M(t_{i+1}^1-t_i^1)\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right).\end{aligned} \tag{C.14}$$

Using equation (C.14) in equation (C.9), when the conditions in equation (26) hold true, we have

$$\begin{aligned}&\lim_{\delta t\to 0}\left|\sum_{i=0}^{N_1-1}W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_{i+1}^1-t_i^1\right)-\sum_{i=0}^{N_3-1}W_2^2\left(\mu\left(t_i^3\right),\hat{\mu}\left(t_i^3\right)\right)\left(t_{i+1}^3-t_i^3\right)\right|\\&\leqslant 2MT\max_i\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right)\to 0.\end{aligned} \tag{C.15}$$

Similarly,

$$\begin{aligned}\lim_{\delta t\to 0}\left|\sum_{i=0}^{N_2-1} W_2^2\left(\mu\left(t_i^2\right),\hat{\mu}\left(t_i^2\right)\right)\left(t_{i+1}^2-t_i^2\right)-\sum_{i=0}^{N_3-1} W_2^2\left(\mu\left(t_i^3\right),\hat{\mu}\left(t_i^3\right)\right)\left(t_{i+1}^3-t_i^3\right)\right| \\ \leqslant 2MT\max_i\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right)\to 0.\end{aligned} \tag{C.16}$$

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

$$\left|\sum_{i=0}^{N_1-1} W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_{i+1}^1-t_i^1\right)-\sum_{i=0}^{N_2-1} W_2^2\left(\mu\left(t_i^2\right),\hat{\mu}\left(t_i^2\right)\right)\left(t_{i+1}^2-t_i^2\right)\right|\to 0, \tag{C.17}$$

which implies the limit

$$\lim_{N\to\infty}\sum_{i=0}^{N-1}\inf_{\pi_i}\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}\left(t_i^1\right)-\hat{\boldsymbol{X}}\left(t_i^1\right)\right|_2^2\right]\left(t_i^1-t_{i-1}^1\right)=\lim_{N\to\infty}\sum_{i=0}^{N-1} W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_i^1-t_{i-1}^1\right) \tag{C.18}$$

exists. From equation (25), the limit

$$\lim_{N\to\infty}\sum_{i=1}^{N-1}\inf_{\pi_i}\mathbb{E}_{\pi_i}\left[\left|\boldsymbol{X}\left(t_i^1\right)-\hat{\boldsymbol{X}}\left(t_i^1\right)\right|_2^2\right]\left(t_i^1-t_{i-1}^1\right)=\tilde{W}_2^2(\mu,\hat{\mu}). \tag{C.19}$$

Specifically, by letting $\max_{i=0}^{n_2-1}(t_{i+1}^2-t_i^2)\to 0$ in equation (C.17), we have

$$\begin{aligned}\left|\sum_{i=0}^{N_1-1} W_2^2\left(\mu\left(t_i^1\right),\hat{\mu}\left(t_i^1\right)\right)\left(t_{i+1}^1-t_i^1\right)-\tilde{W}_2^2(\mu,\hat{\mu})\right| \\ \leqslant 2MT\max_i\left(\sqrt{F_i\Delta t+\Sigma_i+B_i}+\sqrt{\hat{F}_i\Delta t+\hat{\Sigma}_i+\hat{B}_i}\right).\end{aligned} \tag{C.20}$$

This completes the proof of theorem 3.1.

Appendix D. Proof of theorem 3.2

Below, we provide proof for theorem 3.2. First, note that

$$\begin{aligned}\mathbb{E}\left[\left|W_2^2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)-W_2^2\left(\mu_N,\hat{\mu}_N\right)\right|\right]\leqslant\mathbb{E}\left[\left(W_2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)-W_2\left(\mu_N,\hat{\mu}_N\right)\right)^2\right] \\ +2\mathbb{E}\left[\left|W_2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)-W_2\left(\mu_N,\hat{\mu}_N\right)\right|\right]W_2\left(\mu_N,\hat{\mu}_N\right).\end{aligned} \tag{D.1}$$

Using the triangular inequality for the Wasserstein distance [45], we have

$$\begin{aligned}\mathbb{E}\left[\left|W_2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)-W_2\left(\mu_N,\hat{\mu}_N\right)\right|\right]\leqslant\mathbb{E}\left[W_2\left(\mu_N^{\mathrm{e}},\mu_N\right)\right]+\mathbb{E}\left[W_2\left(\hat{\mu}_N^{\mathrm{e}},\hat{\mu}_N\right)\right], \\ \mathbb{E}\left[\left(W_2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)-W_2\left(\mu_N,\hat{\mu}_N\right)\right)^2\right]\leqslant 2\mathbb{E}\left[W_2^2\left(\mu_N^{\mathrm{e}},\mu_N\right)+W_2^2\left(\hat{\mu}_N^{\mathrm{e}},\hat{\mu}_N\right)\right].\end{aligned} \tag{D.2}$$

From theorem 1 in [29], there exists a constant C_0 depending on the dimensionality Nd such that:

$$\begin{aligned}\mathbb{E}\left[W_2^2\left(\mu_N^{\mathrm{e}},\mu_N\right)\right]\leqslant C_0h^2(M_s,Nd)\,\mathbb{E}\left[\sum_{i=0}^{N-1}|\boldsymbol{X}(t_i)|_6^6\Delta t_i^3\right]^{\frac{1}{3}}, \\ \mathbb{E}\left[W_2^2\left(\hat{\mu}_N^{\mathrm{e}},\hat{\mu}_N\right)\right]\leqslant C_0h^2(M_s,Nd)\,\mathbb{E}\left[\sum_{i=0}^{N-1}|\hat{\boldsymbol{X}}(t_i)|_6^6\Delta t_i^3\right]^{\frac{1}{3}},\end{aligned} \tag{D.3}$$

where the function h is defined in equation (36) and $\Delta t_i := (t_{i+1} - t_i), i = 0, \ldots, N-1$. Substituting equations (D.3) and (D.2) into equation (D.1), we conclude that

$$\begin{aligned}
&\mathbb{E}\left[\left|W_2^2\left(\mu_N^{\mathrm{e}}, \mu_N\right) - W_2^2\left(\hat{\mu}_N^{\mathrm{e}}, \hat{\mu}_N^{\mathrm{e}}\right)\right|\right] \\
&\quad \leqslant 2C_0 h^2\left(M_s, Nd\right)\left(\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\boldsymbol{X}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{3}} + \mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{3}}\right) \\
&\qquad + 2\sqrt{C_0} W_2\left(\mu_N, \hat{\mu}_N\right) h\left(M_s, Nd\right)\left(\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\boldsymbol{X}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}} + \mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}}\right)
\end{aligned} \tag{D.4}$$

which proves the inequality (35). Similarly, for each $i = 0, 1, \ldots, N-1$, there exists a constant C_1 depending on the dimensionality d such that

$$\begin{aligned}
&\mathbb{E}\left[\left|W_2^2\left(\mu_N^{\mathrm{e}}(t_i), \hat{\mu}_N^{\mathrm{e}}(t_i)\right) - W_2^2\left(\mu_N(t_i), \hat{\mu}_N(t_i)\right)\right|\right] \Delta t_i \\
&\quad \leqslant 2\sqrt{C_1} h\left(M_s, d\right)\left(\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{6}} + \mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{6}}\right) W_2\left(\mu(t_i), \hat{\mu}(t_i)\right) \Delta t_i \\
&\qquad + 2C_1 h^2\left(M_s, d\right)\left(\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{3}} + \mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{3}}\right) \Delta t_i.
\end{aligned} \tag{D.5}$$

Summing over i in the inequalities (D.5), we find

$$\begin{aligned}
&\mathbb{E}\left[\left|\sum_{i=0}^{N-1}\left(W_2^2\left(\mu_N^{\mathrm{e}}(t_i), \hat{\mu}_N^{\mathrm{e}}(t_i)\right) \Delta t_i - W_2^2\left(\mu_N(t_i), \hat{\mu}_N(t_i)\right) \Delta t_i\right)\right|\right] \\
&\quad \leqslant \sum_{i=0}^{N-1} \mathbb{E}\left[\left|W_2^2\left(\mu_N^{\mathrm{e}}(t_i), \hat{\mu}_N^{\mathrm{e}}(t_i)\right) - W_2^2\left(\mu_N(t_i), \hat{\mu}_N(t_i)\right)\right| \Delta t_i\right] \\
&\quad \leqslant 2\sqrt{C_1} \sum_{i=0}^{N-1}\left(\left(\mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{6}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{6}}\right) W_2\left(\mu_N(t_i), \hat{\mu}_N(t_i)\right) \Delta t_i h\left(M_s, d\right)\right. \\
&\qquad \left. + 2C_1\left(\mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{3}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{3}}\right) \Delta t_i h^2\left(M_s, d\right)\right),
\end{aligned} \tag{D.6}$$

which proves the inequality (37). Furthermore, using the Hölder's inequality, we have

$$\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\boldsymbol{X}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{3}} \cdot \mathbb{E}\left[\sum_{i=0}^{N-1} 1\right]^{\frac{2}{3}} \geqslant \sum_{i=0}^{N-1} \mathbb{E}\left[\left|\boldsymbol{X}(t_i)\right|_6^6\right]^{\frac{1}{3}} \Delta t_i \tag{D.7}$$

and

$$\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{3}} \cdot \mathbb{E}\left[\sum_{i=0}^{N-1} 1\right]^{\frac{2}{3}} \geqslant \sum_{i=0}^{N-1} \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\right]^{\frac{1}{3}} \Delta t_i. \tag{D.8}$$

Furthermore, for any coupled distribution $\pi(\boldsymbol{X}_N, \hat{\boldsymbol{X}}_N)$ whose marginal distributions are μ_N and $\hat{\mu}_N$, we have, by using the Cauchy inequality,

$$\begin{aligned}
&2\sum_{i=0}^{N-1}\left(\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}} + \mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}}\right) \cdot \mathbb{E}_{\pi\left(\boldsymbol{X}_N, \hat{\boldsymbol{X}}_N\right)}\left[\left|\hat{\boldsymbol{X}}(t_i) - \boldsymbol{X}(t_i)\right|_2^2 \Delta t_i\right]^{\frac{1}{2}} \\
&\leqslant 2\mathbb{E}\left[\sum_{i=0}^{N-1} 1\right]^{\frac{1}{3}} \cdot \left(\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}} + \mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6 \Delta t_i^3\right]^{\frac{1}{6}}\right) \\
&\quad \times \mathbb{E}_{\pi\left(\boldsymbol{X}_N, \hat{\boldsymbol{X}}_N\right)}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i) - \boldsymbol{X}(t_i)\right|_2^2 \Delta t_i\right]^{\frac{1}{2}}.
\end{aligned} \tag{D.9}$$

Therefore, by taking the infimum over all coupling distributions $\pi(\boldsymbol{X}_N, \hat{\boldsymbol{X}}_N)$, we conclude that

$$\begin{aligned} &2\sum_{i=0}^{N-1}\left(\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}+\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}\right)W_2\left(\mu\left(t_i\right),\hat{\mu}\left(t_i\right)\right)\sqrt{\Delta t_i} \\ &\quad\leqslant 2\sum_{i=0}^{N-1}\left(\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}+\mathbb{E}\left[\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}\right)\inf_{\pi\left(\boldsymbol{X}_N,\hat{\boldsymbol{X}}_N\right)}\mathbb{E}_{\pi\left(\mu_N,\hat{\mu}_N\right)}\left[\left|\hat{\boldsymbol{X}}(t_i)-\boldsymbol{X}(t_i)\right|_2^2\Delta t_i\right]^{\frac{1}{2}} \\ &\quad\leqslant 2\left(\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}+\mathbb{E}\left[\sum_{i=0}^{N-1}\left|\hat{\boldsymbol{X}}(t_i)\right|_6^6\Delta t_i^3\right]^{\frac{1}{6}}\right)W_2\left(\mu_N,\hat{\mu}_N\right)N^{\frac{1}{3}}. \end{aligned} \tag{D.10}$$

After combining the five inequalities equations (D.7), (D.8), (D.4), (D.6) and (D.10), we conclude that

$$E_1\left(M_s\right)\geqslant CE_2\left(M_s\right)\frac{h\left(M_s,Nd\right)}{h\left(M_s,d\right)}N^{-\frac{2}{3}}, \tag{D.11}$$

where $C:=\frac{1}{\sqrt{10}}\min\left\{\sqrt{\frac{C_0}{C_1}},\frac{C_0}{C_1}\right\}\leqslant\min\left\{\sqrt{\frac{C_0}{C_1}},\frac{C_0}{C_1}\right\}\cdot\min_{x\geqslant 1}\frac{h(x,5)}{h(x,4)}$.

Appendix E. Default training settings

We list the training hyperparameters and gradient descent methods for each example in table E1.

Table E1. Training settings for each example.

Loss	Example 4.1	Example 4.2	Example 4.3
Gradient descent method	AdamW	AdamW	AdamW
Learning rate	0.002	0.003	0.002
Weight decay	0.005	0.02	0.005
No. of epochs	1000	500	400
No. of training trajectories M_s	100	400	300
Hidden layers in Θ_1	2	2	\
Hidden layers in Θ_2	2	2	3
Hidden layers in Θ_3	2	2	3
Activation function	ReLu	ReLu	ReLu
Neurons in each layer in Θ_1	150	150	\
Neurons in each layer in Θ_2	150	150	400
Neurons in each layer in Θ_3	150	150	400
Δt	0.2	0.1	0.2
Number of timesteps N	101	51	51
Initialization	`torch.nn` default	`torch.nn` default	0 for biases $\mathcal{N}(0,10^{-4})$ for weights
Repeat times	10	5	5

Appendix F. Definitions of different loss metrics

Here, we provide definitions of loss functions used in our numerical examples (the definitions of the MSE, mean^2+var, and the MMD loss functions are the same as appendix E in [25]). Since we are using a uniform mesh grid ($t_{i+1}-t_i=\Delta t,\forall i=0,\ldots,N-1$), for simplicity, we shall omit Δt in the calculation of our loss functions:

(i) The squared Wasserstein-2 distance

$$W_2^2\left(\mu_N,\hat{\mu}_N\right)\approx W_2^2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right),$$

where μ_N^{e} and $\hat{\mu}_N^{\mathrm{e}}$ are the empirical distributions of the vector $(\boldsymbol{X}(t_0),\ldots,\boldsymbol{X}(t_{N-1}))$ and $(\hat{\boldsymbol{X}}(t_0),\ldots,\hat{\boldsymbol{X}}(t_{N-1}))$, respectively. In numerical examples, we use the following scaled squared Wasserstein-2 distance:

$$\frac{1}{\Delta t}W_2^2\left(\mu_N^{\mathrm{e}},\hat{\mu}_N^{\mathrm{e}}\right)\approx\texttt{ot.emd2}\left(\frac{1}{M_s}\boldsymbol{I}_{M_s},\frac{1}{M_s}\boldsymbol{I}_{M_s},\boldsymbol{C}\right), \tag{F.1}$$

where `ot.emd2` is the function for solving the earth movers distance problem in the `ot` package of Python [27], M_s is the number of ground truth and predicted trajectories, $\boldsymbol{I}_{M_s}$ is an M_s-dimensional vector whose elements are all 1, and $\boldsymbol{C} \in \mathbb{R}^{M_s \times M_s}$ is a matrix with entries $(\boldsymbol{C})_{ij} = |\boldsymbol{X}_N^i - \hat{\boldsymbol{X}}_N^j|_2^2$. $|\cdot|_2$ is the 2-norm of a vector. $\boldsymbol{X}_N^i$ is the vector of the values of the i^{th} ground truth trajectory at time points $t_0, \ldots, t_{N-1}$, and $\hat{\boldsymbol{X}}_N^j$ is the vector of the values of the j^{th} predicted trajectory at time points $t_0, \ldots, t_{N-1}$.

(ii) The temporally decoupled squared Wasserstein-2 distance (equation (40)). In numerical examples, we use the following scaled temporally decoupled squared Wasserstein-2 distance:

$$\frac{1}{\Delta t}\tilde{W}_2^2(\mu_N, \hat{\mu}_N) \approx \sum_{i=1}^{N-1} W_2^2\left(\mu^{\mathrm{e}}(t_i), \hat{\mu}^{\mathrm{e}}(t_i)\right),$$

where Δt is the time step and W_2 is the Wasserstein-2 distance between two empirical distributions of $\boldsymbol{X}(t_i)$ and $\hat{\boldsymbol{X}}(t_i)$, denoted by $\mu^{\mathrm{e}}(t_i), \hat{\mu}^{\mathrm{e}}(t_i)$, respectively. These distributions are calculated by the samples of the trajectories of $\boldsymbol{X}(t), \hat{\boldsymbol{X}}(t)$ at a given time step $t = t_i$, respectively. $W_2^2(\mu_N^{\mathrm{e}}(t_i), \hat{\mu}_N^{\mathrm{e}}(t_i))$ is calculated using the `ot.emd2` function, i.e.

$$W_2^2\left(\mu^{\mathrm{e}}(t_i), \hat{\mu}^{\mathrm{e}}(t_i)\right) \approx \texttt{ot.emd2}\left(\frac{1}{M_s}\boldsymbol{I}_{M_s}, \frac{1}{M_s}\boldsymbol{I}_{M_s}, \boldsymbol{C}(t_i)\right), \tag{F.2}$$

where $\boldsymbol{I}_{M_s}$ is an M_s-dimensional vector whose elements are all 1, and $\boldsymbol{C} \in \mathbb{R}^{M_s \times M_s}$ is a matrix with entries $(\boldsymbol{C})_{sj} = |\boldsymbol{X}^s(t_i) - \hat{\boldsymbol{X}}^j(t_i)|_2^2$. $\boldsymbol{X}^s(t_i)$ is the vector of values of the s^{th} ground truth trajectory at time t_i and $\hat{\boldsymbol{X}}^s(t_i)$ is the vector of values of the s^{th} trajectory generated by the reconstructed jump-diffusion process at time t_i.

(iii) The Wasserstein-1 distance

$$W_1(\mu_N, \hat{\mu}_N) \approx W_1\left(\mu_N^{\mathrm{e}}, \hat{\mu}_N^{\mathrm{e}}\right),$$

where μ_N^{e} and $\hat{\mu}_N^{\mathrm{e}}$ are the empirical distributions of the vector $(\boldsymbol{X}(t_0), \ldots, \boldsymbol{X}(t_{N-1}))$ and $(\hat{\boldsymbol{X}}(t_0), \ldots, \hat{\boldsymbol{X}}(t_{N-1}))$, respectively. In numerical examples, we use the following scaled W_1 distance:

$$\frac{1}{\Delta t}W_1\left(\mu_N^{\mathrm{e}}, \hat{\mu}_N^{\mathrm{e}}\right) \approx \texttt{ot.emd2}\left(\frac{1}{M_s}\boldsymbol{I}_{M_s}, \frac{1}{M_s}\boldsymbol{I}_{M_s}, \boldsymbol{C}\right), \tag{F.3}$$

where `ot.emd2` is the function for solving the earth movers distance problem in the `ot` package of Python, M_s is the number of ground truth and predicted trajectories, $\boldsymbol{I}_{M_s}$ is an M_s-dimensional vector whose elements are all 1, and $\boldsymbol{C} \in \mathbb{R}^{M_s \times M_s}$ is a matrix with entries $(\boldsymbol{C})_{ij} = |\boldsymbol{X}_N^i - \hat{\boldsymbol{X}}_N^j|_2$. $\boldsymbol{X}_N^i$ is the vector of the values of the ith ground-truth trajectory at time points $t_0, \ldots, t_{N-1}$, and $\hat{\boldsymbol{X}}_N^j$ is the vector of the values of the jth predicted trajectory at time points $t_0, \ldots, t_{N-1}$.

(iv) Mean squared error (MSE) between the trajectories, where M_s is the total number of the ground truth and predicted trajectories. $\boldsymbol{X}_{i,j}$ and $\hat{\boldsymbol{X}}_{i,j}$ are the values of the jth ground-truth and prediction trajectories at time t_i, respectively:

$$\mathrm{MSE}\left(\boldsymbol{X}, \widehat{\boldsymbol{X}}\right) = \frac{1}{M_s N}\sum_{i=0}^{N-1}\sum_{j=1}^{M_s}\left(\boldsymbol{X}_{i,j} - \hat{\boldsymbol{X}}_{i,j}\right)^2.$$

(v) The summation of squared distance between mean trajectories and absolute values of the discrepancies in variances of trajectories, which is a common practice for estimating the parameters of an SDE. We shall denote this loss function by

$$\left(\mathrm{mean}^2 + \mathrm{var}\right)\left(\boldsymbol{X}, \hat{\boldsymbol{X}}\right) = \sum_{i=0}^{N-1}\left[\left(\frac{1}{M_s}\sum_{j=1}^{M_s}\left(\boldsymbol{X}_{i,j} - \hat{\boldsymbol{X}}_{i,j}\right)\right)^2 + \left|\mathrm{var}\left(\boldsymbol{X}_i\right) - \mathrm{var}\left(\hat{\boldsymbol{X}}_i\right)\right|\right].$$

Here M_s and $\boldsymbol{X}_{i,j}$ and $\hat{\boldsymbol{X}}_{i,j}$ have the same meaning as in the MSE definition. $\mathrm{var}(\boldsymbol{X}_i)$ and $\mathrm{var}(\hat{\boldsymbol{X}}_i)$ are the variances of the empirical distributions of $\boldsymbol{X}(t_i), \hat{\boldsymbol{X}}(t_i)$, respectively.

(vi) MMD (maximum mean discrepancy) In our numerical examples, we use the following MMD loss function [46]:

$$\mathrm{MMD}\left(X, \hat{X}\right) = \sum_{i=1}^{N-1}\left(\mathbb{E}\left[K\left(\boldsymbol{X}_i, \boldsymbol{X}_i\right)\right] - 2\mathbb{E}\left[K\left(\boldsymbol{X}_i, \hat{\boldsymbol{X}}_i\right)\right] + \mathbb{E}\left[K\left(\hat{\boldsymbol{X}}_i, \hat{\boldsymbol{X}}_i\right)\right]\right),$$

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

Appendix G. Changing the number of training trajectories for different loss functions

Here, we consider the reconstruction of the jump-diffusion process given in equation (57) as a function of the number of trajectories and the loss functions used for training. We generate trajectories from the ground-truth jump-diffusion process equation (57) with $b = 4, a = -1, \sigma_0 = 0.4, y_0 = 1, T = 20.2$ and the initial condition $X_0 = 2$. Except for the number of training samples M_s, the training setting and hyperparameters are the same as those described in table E1 for Example 4.1. The reconstruction accuracy of the drift, diffusion, and jump functions for all methods tends to improve with an increasing number of training trajectories. Additionally, we find that our proposed temporally decoupled squared W_2 method usually gives more accurate reconstructed drift, diffusion, and jump functions compared to using other loss functions or methods. Using the MMD loss function could yield more accurate reconstructed jump functions when the number of training samples is small ($\leqslant 64$). However, the reconstruction of the drift and diffusion functions when using the MMD loss function is not as good as that of using our temporally decoupled squared W_2 method. The results are plotted in figure G1.

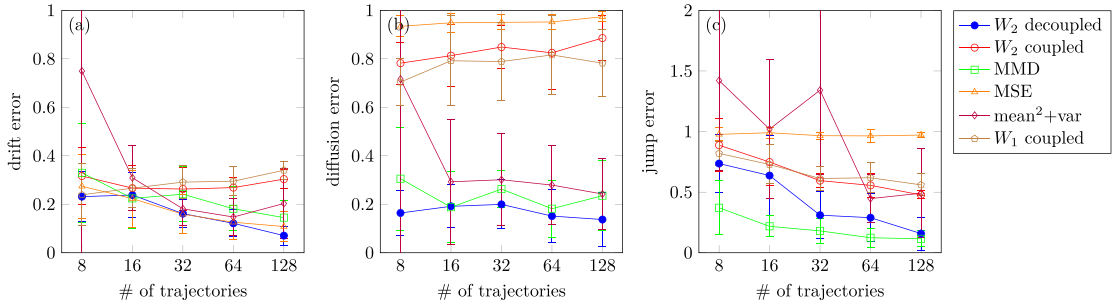

Figure G1. The error in the reconstructed drift, diffusion, and jump functions of the 1D jump-diffusion process equation (57) for different loss functions. The number of training trajectories is varied as different loss functions are used.

Appendix H. Varying the coefficients that determine diffusion and jump functions

Here, we consider changing the two parameters σ_0, y_0 in equation (57) of example 4.1. With larger σ_0, y_0, the trajectories generated by equation (57) will be subject to greater fluctuations. We use the temporally squared W_2 distance as the loss function. We vary σ_0 to range from 0.2 to 0.4 and vary y_0 from 0.5 to 1. We repeat our experiments 10 times, and we plot the temporally squared W_2 distance as well as the errors of the reconstructed $\hat{f}, \hat{\sigma}, \hat{\beta}$.

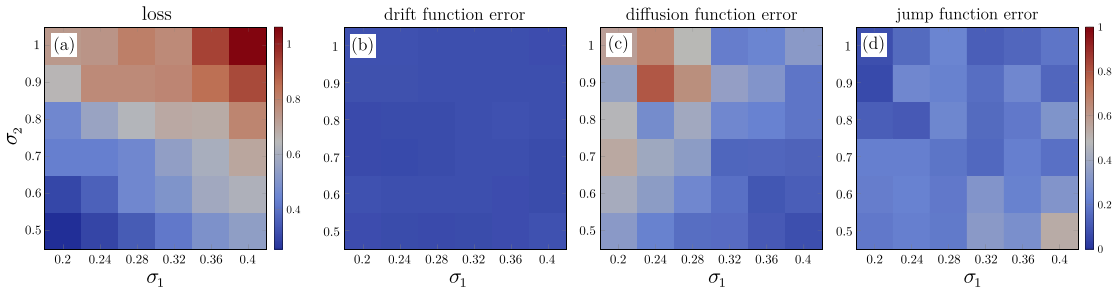

Figure H1. (a) The temporally decoupled squared Wasserstein distance $\tilde{W}_2^2(\mu_N, \hat{\mu}_N)$. (b) the average relative errors in the reconstructed drift function $\hat{f}(x)$; (c) the average relative errors in the reconstructed diffusion function $\hat{\sigma}(x)$; (d) the average relative errors in the reconstructed jump functions $\hat{\beta}(x)$.

From figure H1(a), larger σ_0, y_0 lead to larger $\tilde{W}_2^2(\mu_N, \hat{\mu}_N)$. This could be because larger σ_0, y_0 lead to ground truth trajectories with larger fluctuations, rendering the underlying dynamics harder to reconstruct. Figure H1(b) implies that the drift function can be accurately reconstructed and is insensitive to different σ_0, y_0. As seen in Figure H1(c), if σ_0 is small, the relative error in the reconstructed diffusion function can be well controlled around 0.1; when σ_0 is larger, it is harder to reconstruct the diffusion function and the relative error in the reconstructed diffusion function $\hat{\sigma}$ will be larger. Figure H1(d) shows that the reconstruction of the jump function $\hat{\beta}$ is not very sensitive to different values of σ_0 and y_0.

Appendix I. Reconstructing equation (59) in example 4.2 with different numbers of trajectories in the training set

Here, we carry out an additional numerical experiment of reconstructing equation (59) by changing the number of trajectories in the training set. We define the ground truth jump-diffusion process by the drift function $\alpha(X,t) := r$, and the diffusion function and jump functions $\sigma(X,t) = \beta(X,t) = 0.1\sqrt{|X|}$. We consider four scenarios: i) provide no prior information and reconstruct drift, diffusion, and jump functions, ii) provide the drift function $\alpha(X,t)$ as prior information and reconstruct the diffusion and jump functions, iii) provide the diffusion function $\sigma(X,t)$ as prior information and reconstruct the drift and jump functions, and iv) provide the jump function $\beta(X,t)$ as prior information and reconstruct the drift and diffusion functions.

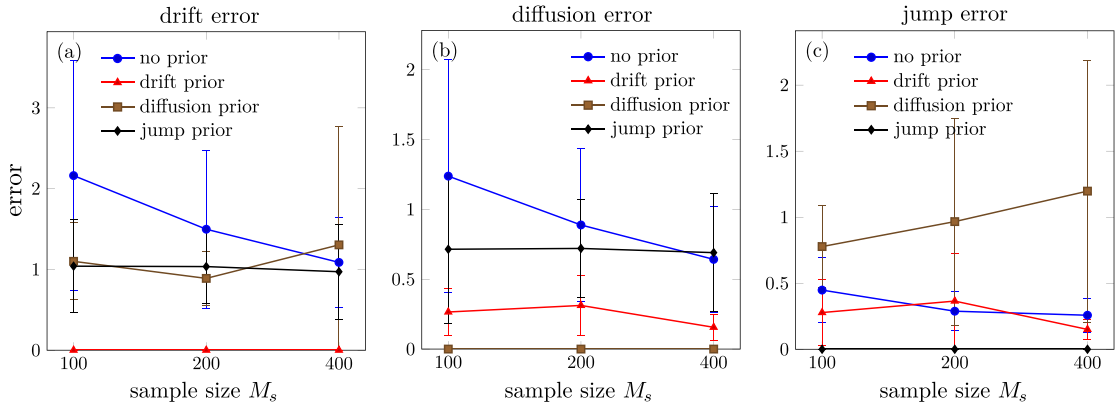

Figure I1. The reconstruction errors in the drift, diffusion, and jump functions defined in equations (54) and (55) as a function of the number of trajectories M_s when different prior information is provided. The results are averaged over 5 independent experiments. Training hyperparameters are the same as those used in example 4.2 listed in table E1.

As seen in figures I1(b) and (c), providing the drift function as prior information greatly boosts the efficiency of our temporally decoupled squared W_2 method allowing it to accurately reconstructing the unknown diffusion and jump functions even with as few as 100 trajectories for training. Also, even with no prior information, the errors in the reconstructed drift, diffusion, or jump function decrease when the number of trajectories in the training set increases (figures I1(a)–(c)). This indicates that even without prior information, our temporally decoupled squared W_2 method can accurately reconstruct equation (59) when provided a sufficient number of training trajectories.

When the diffusion or jump function is given as prior information, the errors of the reconstructed unknown functions do not decrease much as the number of trajectories for training M_s increases. Even with the correct diffusion or jump function, different realizations of the Brownian motion or the compensated Poisson process yield very different trajectories so that providing the diffusion or jump function may provide little information in discriminating trajectories.

Appendix J. Reconstructing equation (59) in example 4.2 with different parameters in the diffusion and jump functions when providing the drift function

Here, given the drift function $\alpha(X,t) := r$, we carry out an additional numerical experiment of reconstructing equation (59) by varying the parameters σ_0, β_0 that determine the strength of the Brownian-type and compensated-Poisson-type noise in equations (60)–(62).

Figure J1 shows our temporally decoupled squared W_2-distance loss function can be used to accurately reconstruct the diffusion function and the jump function $\sigma(X,t), \beta(X,t)$ in equation (59), even when

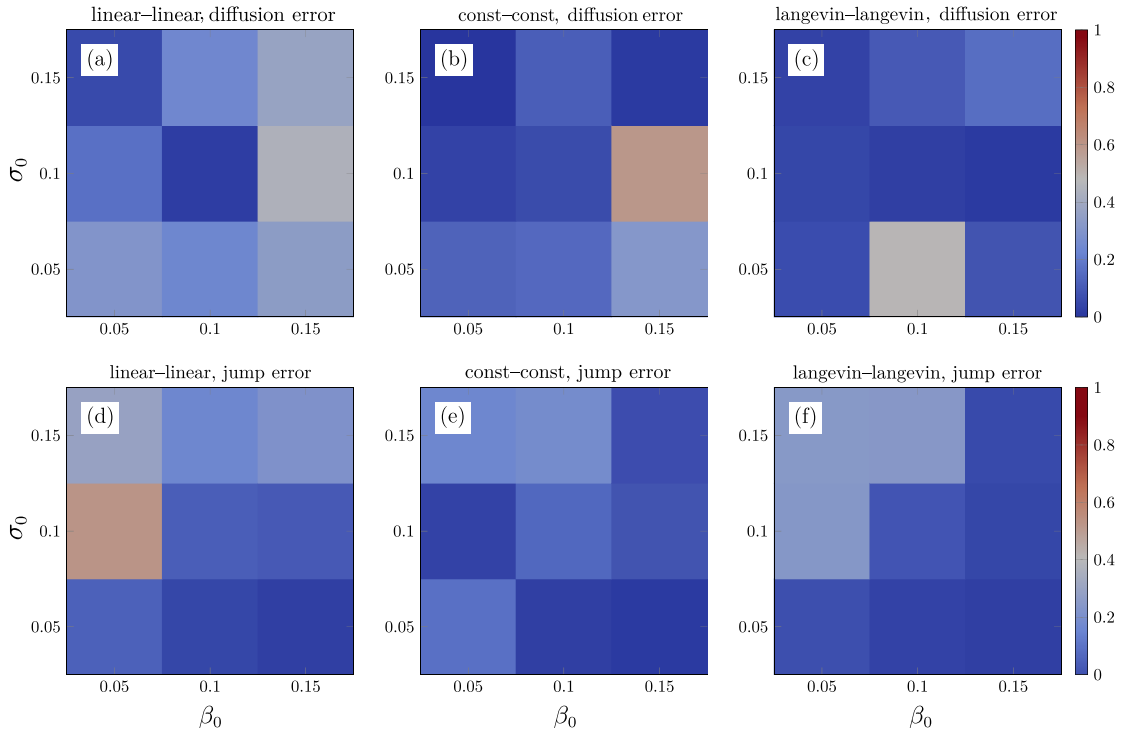

Figure J1. The reconstruction errors in the diffusion, and jump functions defined in equations (54) and (55) w.r.t. the two parameters that determine the strength of noise σ_0 and β_0 in equations (60)–(62). Here, const–const indicates we are using equation (60) for both the diffusion and jump functions; linear–linear indicates we are using equation (61) for both the diffusion and jump functions; langevin–langevin indicates we are using equation (62) for both the diffusion and jump functions. The results are averaged over 5 independent experiments. Training hyperparameters are the same as example 4.2 in table E1.

different forms of $\sigma(X,t), \beta(X,t)$ in equations (60)–(62) and different noise strengths σ_0, β_0 are given. The average errors (averaged over all choices of σ_0, β_0) in the reconstructed diffusion function $\hat{\sigma}$ are 0.171 (const–const), 0.217 (linear–linear), and 0.176 (langevin–langevin). The average errors (averaged over all choices of σ_0, β_0) in the reconstructed jump function $\hat{\sigma}$ are 0.173 (const–const), 0.188 (linear–linear), and 0.184 (langevin–langevin).

Appendix K. Neural network architecture

Here, we investigate how the neural network architecture, i.e. the number of hidden layers and the number of neurons in each layer, influence the accuracy of reconstructing the 2D jump-diffusion process (equation (63)). We vary only the number of hidden layers and the number of neurons per layer for the parameterized neural networks that we use to approximate the diffusion and jump functions $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$ in equation (63). We set the parameters to be $c_1 = -0.5, c_2 = -1$ and $\sigma_0 = \beta_0 = 0.1$ in equations (65) and (66), and consider 200 trajectories.

From table K1, we see that increasing the number of hidden layers and increasing the number of neurons per hidden layer can both increase the accuracy of the reconstructed $\hat{\boldsymbol{\sigma}}$ and $\hat{\boldsymbol{\beta}}$. However, with a fixed number of neurons per hidden layer (200), when the number of hidden layers in the feed-forward neural network is greater than 3, the errors in the reconstructed $\boldsymbol{\sigma}$ and $\boldsymbol{\beta}$ increase. This behavior may be due to vanishing gradients during training of deep neural networks [47]; in this case, the ResNet technique [48] can be considered if deep neural networks are used. On the other hand, using a deeper or wider network requires more memory usage and longer run times. For reconstructing equation (63), we find an optimal neural network architecture consisting of about three hidden layers containing ∼400 neurons each.

We found that adding Dropout layers did not lead to improved reconstruction accuracy. The underlying reason could be that the diffusion and jump functions of the jump-diffusion process to be reconstructed equation (63) are deterministic, while the dropout layer will randomly select neurons in the hidden layers to ignore. Such randomness for the neural-network-parameterized diffusion and jump function induced by the Dropout layers is not compatible with the deterministic diffusion and jump functions to be reconstructed and introduces new stochasticity on top of intrinsic noise in the jump-diffusion process. Therefore, applying the Dropout technique will not improve the reconstruction accuracy of the diffusion and jump functions.

Table K1. Reconstructing the jump-diffusion process equation (63) when using neural networks with different numbers of hidden layers and neurons per layer to parameterize $\hat{f}, \hat{\sigma}$. We also applied Dropout with dropout probability p. Other training hyperparameters are the same as those used in table E1 of example 4.3.

Width	Layer	Relative Errors in $\hat{\sigma}$	Relative Errors in $\hat{\beta}$	N_{repeats}
25	3	0.6836($\pm$0.5177)	0.5554($\pm$0.4024)	5
50	3	0.8051($\pm$0.4756)	0.7413($\pm$0.3515)	5
100	3	0.6376($\pm$0.3261)	0.5085($\pm$0.2841)	5
200	3	0.6101($\pm$0.2435)	0.5280($\pm$0.2038)	5
400	3	0.2619($\pm$0.1859)	0.2837($\pm$0.1961)	5
200	1	0.7143($\pm$0.8451)	0.6178($\pm$0.2925)	5
200	2	0.6984($\pm$0.4989)	0.6326($\pm$0.4445)	5
200	4	0.7605($\pm$0.3837)	0.6750($\pm$0.2761)	5
400	3 (Dropout $p = 0.05$)	0.6574($\pm$0.0674)	0.6162($\pm$0.0647)	5
400	3 (Dropout $p = 0.1$)	0.6083($\pm$0.1240)	0.5324($\pm$0.0728)	5
400	3 (Dropout $p = 0.2$)	0.6250($\pm$0.1488)	0.5392($\pm$0.0829)	5

How optimal architectures evolve when reconstructing different multidimensional jump-diffusion processes requires further exploration.

ORCID iDs

Mingtao Xia https://orcid.org/0000-0002-2116-4712
Tom Chou https://orcid.org/0000-0003-0785-6349

References

[1] Merton R C 1976 Option pricing when underlying stock returns are discontinuous *J. Financ. Econ.* **3** 125–44
[2] Jang J 2007 Jump diffusion processes and their applications in insurance and finance *Insur. Math. Econ.* **41** 62–70
[3] Maekawa K, Lee S, Morimoto T and Kawai K-ichi 2008 Jump diffusion model with application to the Japanese stock market *Math. Comput. Simul.* **78** 223–36
[4] Gao J-X, Wang Z-Y, Zhang M Q, Qian M-P and Jiang D-Q 2022 A data-driven method to learn a jump diffusion process from aggregate biological gene expression data *J. Theor. Biol.* **532** 110923
[5] Tesfay A, Saeed T, Zeb A, Tesfay D, Khalaf A and Brannan J 2021 Dynamics of a stochastic COVID-19 epidemic model with jump-diffusion *Adv. Differ. Equ.* **2021** 1–19
[6] Mehri S and Scheutzow M 2019 A stochastic Gronwall lemma and well-posedness of path-dependent SDEs driven by martingale noise (arXiv: 1908.10646)
[7] Casella B and Roberts G O 2011 Exact simulation of jump-diffusion processes with Monte Carlo applications *Methodol. Comput. Appl. Probab.* **13** 449–73
[8] Metwally S A K and Atiya A F 2002 Using Brownian bridge for fast simulation of jump-diffusion processes and barrier options *J. Derivatives* **10** 43–54
[9] Kou S G and Wang H 2003 First passage times of a jump diffusion process *Adv. Appl. Probab.* **35** 504–31
[10] Zhang D and Melnik R V N 2009 First passage time for multivariate jump-diffusion processes in finance and other areas of applications *Appl. Stoch. Models Bus. Ind.* **25** 565–82
[11] Mies F, Sadr M and Torrilhon M 2023 An efficient jump-diffusion approximation of the Boltzmann equation *J. Comput. Phys.* **490** 112308
[12] Dubey V, Dueby S and Daschakraborty S 2021 Breakdown of the Stokes-Einstein relation in supercooled water: the jump-diffusion perspective *Phys. Chem. Chem. Phys.* **23** 19964–86
[13] Barchielli A and Pellegrini C 2010 Jump-diffusion unraveling of a non-Markovian generalized Lindblad master equation *J. Math. Phys.* **51** 112104
[14] Chaintron L-P, Kimmig F, Caruel M and Moireau P 2023 A jump-diffusion stochastic formalism for muscle contraction models at multiple timescales *J. Appl. Phys.* **134** 194901
[15] Rydin Gorjão L, Heysel J, Lehnertz K and Reza Rahimi Tabar M 2019 Analysis and data-driven reconstruction of bivariate jump-diffusion processes *Phys. Rev.* E **100** 062127
[16] Ramezani C A and Zeng Y 1998 Maximum likelihood estimation of asymmetric jump-diffusion processes: application to security prices available at SSRN 606361 (https://doi.org/10.2139/ssrn.606361)
[17] Rydin Gorjão L, Witthaut D and Lind P G 2023 jumpdiff: a python library for statistical inference of jump-diffusion processes in observational or experimental data sets *J. Stat. Softw.* **105** 1–22
[18] Li X, Leonard Wong T-K, Chen R T Q and Duvenaud D 2020 Scalable gradients for stochastic differential equations *Int. Conf. on Artificial Intelligence and Statistics* (PMLR), pp 3870–82
[19] Tzen B and Raginsky M 2019 Neural stochastic differential equations: deep latent Gaussian models in the diffusion limit (arXiv: 1905.09883)
[20] Tong A, Nguyen-Tang T, Tran T and Choi J 2022 Learning fractional white noises in neural stochastic differential equations *Advances in Neural Information Processing Systems* vol 35 pp 37660–75
[21] Kidger P, Foster J, Xuechen Li and Lyons T J 2021 Neural SDEs as infinite-dimensional GANs *Int. Conf. on Machine Learning* (PMLR) pp 5453–63
[22] Chen Y and Xiu D 2023 Learning stochastic dynamical system via flow map operator (arXiv: 2305.03874)
[23] Villani C *et al* 2009 *Optimal Transport: Old and New* vol 338 (Springer) (https://doi.org/10.1007/978-3-540-71050-9)

[24] Zheng W, Wang F-Y and Gou C 2020 Nonparametric different-feature selection using Wasserstein distance *2020 IEEE 32nd Int. Conf. on Tools With Artificial Intelligence (ICTAI)* (IEEE) pp 982–8
[25] Xia M, Li X, Shen Q and Chou T 2024 Squared Wasserstein-2 distance for efficient reconstruction of stochastic differential equations (arXiv: 2401.11354)
[26] Breton J-C and Privault N 2024 Wasserstein distance estimates for jump-diffusion processes *Stoch. Process. Appl.* **172** 104334
[27] Flamary R *et al* 2021 POT: Python optimal transport *J. Mach. Learn. Res.* **22** 1–8
[28] Arras B and Houdré C 2019 *On Stein's Method for Infinitely Divisible Laws With Finite First Moment* (Springer)
[29] Fournier N and Guillin A 2015 On the rate of convergence in Wasserstein distance of the empirical measure *Probab. Theory Relat. Fields* **162** 707–38
[30] Dowson D C and Landau B V 1982 The Fréchet distance between multivariate normal distributions *J. Multivariate Anal.* **12** 450–5
[31] Hull J 1993 *Options, Futures and Other Derivative Securities* vol 7 (Prentice Hall)
[32] Koudriavtsev A B, Jameson R F and Linert W 2011 *The law of Mass Action* (Springer)
[33] Chellaboina V, Bhat S P, Haddad W M and Bernstein D S 2009 Modeling and analysis of mass-action kinetics *IEEE Control Syst. Mag.* **29** 60–78
[34] Shaojun M, Liu S, Zha H and Zhou H 2021 Learning stochastic behaviour from aggregate data *Int. Conf. on Machine Learning* (PMLR) pp 7258–67
[35] Hinton G E 2012 Improving neural networks by preventing co-adaptation of feature detectors (arXiv: 1207.0580)
[36] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15** 1929–58
[37] Mischler S, Mouhot C and Wennberg B 2015 A new approach to quantitative propagation of chaos for drift, diffusion and jump processes *Probab. Theory Relat. Fields* **161** 1–59
[38] Ye X and Wang X 2023 Hidden oscillation and chaotic sea in a novel 3D chaotic system with exponential function *Nonlinear Dyn.* **111** 15477–86
[39] Bertoin J 1996 *Lévy Processes* vol 121 (Cambridge University Press)
[40] Barndorff-Nielsen O E, Mikosch T and Resnick S I 2001 *Lévy Processes: Theory and Applications* (Springer)
[41] Arya G, Schauer M, Schäfer F and Rackauckas C 2022 Automatic differentiation of programs with discrete randomness *Advances in Neural Information Processing Systems* vol 35 pp 10435–47
[42] Buonocore S and Sen M 2021 Anomalous diffusion of cosmic rays: a geometric approach *AIP Adv.* **11** 055221
[43] Kumar A, Barda H, Klinger L, Finnis M W, Lordi V, Rabkin E and Srolovitz D J 2018 Anomalous diffusion along metal/ceramic interfaces *Nat. Commun.* **9** 5251
[44] Jiang Q and Wan L 2024 A physics-informed neural SDE network for learning cellular dynamics from time-series scRNA-seq data *Bioinformatics* **40** ii120–ii127
[45] Clement P and Desch W 2008 An elementary proof of the triangle inequality for the Wasserstein metric *Proc. American Mathematical Society* vol 136 pp 333–9
[46] Li Y, Swersky K and Zemel R 2015 Generative moment matching networks *Int. Conf. on Machine Learning* (PMLR) pp 1718–27
[47] Glorot X and Bengio Y 2010 Understanding the difficulty of training deep feedforward neural networks *Proc. 13th Int. Conf. on Artificial Intelligence and Statistics* (*JMLR Workshop and Conf. Proc.*), pp 249–56
[48] Kaiming H, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 770–8