Monte Carlo Simulation of Stochastic Differential Equation to Study Information Geometry

Thiruthummal, A. A. & Kim, E

DOI 10.3390/e24081113
ISSN 1099-4300

Publisher: MDPI

entropy

MDPI

Article

Monte Carlo Simulation of Stochastic Differential Equation to Study Information Geometry

Abhiram Anand Thiruthummal * and Eun-jin Kim

Centre for Fluid and Complex Systems, Coventry University, Coventry CV1 5FB, UK
* Correspondence: thiruthuma@uni.coventry.ac.uk

Abstract: Information Geometry is a useful tool to study and compare the solutions of a Stochastic Differential Equations (SDEs) for non-equilibrium systems. As an alternative method to solving the Fokker–Planck equation, we propose a new method to calculate time-dependent probability density functions (PDFs) and to study Information Geometry using Monte Carlo (MC) simulation of SDEs. Specifically, we develop a new MC SDE method to overcome the challenges in calculating a time-dependent PDF and information geometric diagnostics and to speed up simulations by utilizing GPU computing. Using MC SDE simulations, we reproduce Information Geometric scaling relations found from the Fokker–Planck method for the case of a stochastic process with linear and cubic damping terms. We showcase the advantage of MC SDE simulation over FPE solvers by calculating unequal time joint PDFs. For the linear process with a linear damping force, joint PDF is found to be a Gaussian. In contrast, for the cubic process with a cubic damping force, joint PDF exhibits a bimodal structure, even in a stationary state. This suggests a finite memory time induced by a nonlinear force. Furthermore, several power-law scalings in the characteristics of bimodal PDFs are identified and investigated.

Keywords: information geometry; information length; stochastic differential equation; Langevin equation; Monte Carlo; GPU; simulation; Fokker–Planck equation; Milstein; non-linear SDE

Citation: Thiruthummal, A.A.; Kim, E.-j. Monte Carlo Simulation of Stochastic Differential Equation to Study Information Geometry. *Entropy* **2022**, *24*, 1113. https://doi.org/10.3390/e24081113

Academic Editor: Frank Nielsen

Received: 27 June 2022
Accepted: 9 August 2022
Published: 12 August 2022

1. Introduction

Stochastic Differential Equations (SDEs) (Equation (5)) are used to model various phenomena in nature, including Brownian motion, asset pricing, population dynamics, COVID-19 spread and interaction [1–6], and various other non-equilibrium processes. Due to their stochasticity, SDEs do not have an unique solution, but a distribution of solutions. A Fokker–Planck Equation (FPE) [7] is a Partial Differential Equation (PDE) that describes how the probability density of solutions of a SDE evolves with time.

Comparing solutions of different SDEs can be achieved by looking at different statistics of the solutions like mean and variance. However, when we are interested in large fluctuations and extreme events in the solutions, simple statistics might not suffice. In such cases, quantifying and comparing the time evolution of probability density functions (PDFs) of solutions will provide us with more information [8]. The time evolution of PDFs can be studied and compared through the framework of information geometry [9], wherein PDFs are considered as points on a Riemannian manifold and their time evolution can be considered as a motion on this manifold. In general, in order to have a manifold structure on the probability space in information geometry, we need to define a metric. Several different metrics can be defined on a probability space [10–13].

Different metrics have different physical and mathematical significance. For example, the Wasserstein metric (also known as the Earth mover's distance) naturally comes up in optimal transport problems [14]; the Ruppeiner metric is based on the geometry of equilibrium thermodynamics [13]. In this work, we use a metric related to the Fisher Information [15], known as the Fisher Information metric [16,17].

$$g_{jk}(\theta) := \int_X \frac{\partial \log p(x;\{\theta\})}{\partial \theta_j} \frac{\partial \log p(x;\{\theta\})}{\partial \theta_k} p(x;\{\theta\}) dx \tag{1}$$

Here, $p(x;\{\theta\})$ denotes a continuous family of PDFs parametrized by parameters $\{\theta\}$. This metric was used to physically represent the number of statistically distinguishable states [11,18,19]. Note that two Gaussians with same standard deviation but different means are statistically indistinguishable, if the difference in their means are much smaller than their standard deviation.

If a time-dependent PDF $p(x,t)$ is considered as a continuous family of PDFs parameterized by a single parameter, time t, the scalar metric can be given by:

$$g(t) = \int d\mathbf{x} \frac{1}{p(\mathbf{x},t)} \left[\frac{\partial p(\mathbf{x},t)}{\partial t}\right]^2 \tag{2}$$

However, time in classical mechanics is a passive quantity that cannot be changed by an external control. The infinitesimal distance $d\mathcal{L}$ on the manifold is then given by $d\mathcal{L}^2 = g(t)dt^2$. Here, $\mathcal{L}$ is the Information Length defined by:

$$\mathcal{L}(t) := \int_0^t dt_1 \sqrt{\int d\mathbf{x} \frac{1}{p(\mathbf{x},t_1)} \left[\frac{\partial p(\mathbf{x},t_1)}{\partial t_1}\right]^2} \tag{3}$$

The Information Length $\mathcal{L}$ represents the dimensionless distance, which measures the total distance traveled on the manifold, or the total number of distinguishable states a system passes through during the course of its evolution. It has previously been used in optimization problems [20]. It was also used to study dynamical systems, thermodynamics, phase transitions, memory effects, and self-organization [21–28].

The gradient of $\mathcal{L}$, $\lim_{dt\to 0} d\mathcal{L}/dt \equiv \Gamma$, then represents a velocity on this manifold

$$\Gamma(t) := \sqrt{\int d\mathbf{x} \frac{1}{p(\mathbf{x},t)} \left[\frac{\partial p(\mathbf{x},t)}{\partial t}\right]^2} \tag{4}$$

Note that we use the notation Γ instead of $\sqrt{g}$ to make it clear that it is a quantity defined for a time-dependent PDF. Γ represents the rate of change of statistically distinguishable states in a time-evolving PDF, and is sometimes referred to in the literature [29–31] as the Information Rate. Note that in information theory [32], the term "Information Rate" is used for the rate at which information is passed over the channel [33].

As for the physical significance of Γ, in a non-equilibrium thermodynamic system, Γ is related to the entropy production [30]. Γ has also been used to study causality [29] and abrupt changes in the dynamics of a system [8]. Γ^2 is equivalent to the (symmetric) KL divergence of infinitesimally close PDFs, as shown in Appendix E. It should also be noted that $\Gamma(t)$ defined by Equation (4) has the dimensions of t^{-1}, and the time-integral of $\Gamma(t)$ gives a dimensionless distance $\mathcal{L}$ in Equation (3).

Due to the lack of general mathematical techniques to solve SDEs or its associated FPE, analytical study of SDEs using Information Geometry (Γ and $\mathcal{L}$) has been limited to a few special cases [24,25,34,35]. To date, numerical studies have relied on solving the associated FPE [24,34], which has the advantage of generating smooth time-dependent PDFs and information diagnostics, but has the limitations outlined in Table 1. To overcome the limitations of a FPE solver, in this work, we develop a Monte Carlo (MC) method to study time-dependent PDFs and the Information Geometry of SDEs.

Table 1. Comparison between grid-based FPE solver and MC SDE simulation.

	Grid-Based FPE Solver	**MC SDE Simulation**
Accuracy	Depends on grid size. No well-defined prescription on choosing grid-size.	Depends on number of samples(n) [36]. Typically less accurate for practical sample sizes.
Boundary condition	Requires carefully chosen non-trivial boundary conditions. Cannot handle discontinuous initial conditions such as Dirac delta function.	Requires only an initial distribution as boundary condition.
Memory Usage & Runtime	Scales exponentially with dimension d. $\mathcal{O}(n_1 n_2 \dots n_d)$. Here, $n_1, n_2, \dots, n_d$ are the number of grid points along each dimension.	Scales linearly with dimension d. $\mathcal{O}(nd)$. Here, n is the number of samples.
Correlation Study	Cannot study correlations and associated memory effects using FPE.	Can study correlations. See Section 4 for unequal time joint PDF estimates.

The main aims of this paper are twofold. The first aim is to develop a new MC SDE simulation method and validate it by recovering the previous results obtained using the FPE method. The second is to calculate unequal time joint PDFs and investigate the effect of nonlinear forces on PDF form and various power-scaling relations. The remainder of the paper is organized as follows: Section 2.1 gives a brief introduction of the theory of MC SDE simulation. Section 2.2 develops the methods to measure Information Geometry from the simulation. Using this method, we compare a linear and a nonlinear SDE in Section 3. In Section 4, we showcase the measurement of joint PDFs of the same variable but at unequal times, which is not possible using FPE solvers. We then study a type of phase transition in the joint PDF of the nonlinear SDE and numerically verify its theoretically calculated scaling relations. Discussions are found in Section 5.

2. Methods

2.1. SDE Simulation

A typical SDE for the variable x in d dimensions has the following form:

$$d\vec{\mathbf{x}}_t = \vec{\boldsymbol{\mu}}(\vec{\mathbf{x}}_t, t)\, dt + \boldsymbol{\sigma}(\vec{\mathbf{x}}_t, t).d\vec{\mathbf{W}}_t \tag{5}$$

Here, $\vec{\boldsymbol{\mu}}$ is known as the drift vector (drift coefficient in 1D), $\mathbf{D} := \boldsymbol{\sigma}.\boldsymbol{\sigma}^T/2$ the diffusion tensor (diffusion coefficient in 1D) and $\vec{\mathbf{W}}_t$ is the Weiner process [37]. $d\vec{\mathbf{W}}_t$ represents an infinitesimal random noise term, making the equation stochastic. Generalizations to SDEs can be achieved with more general noise terms and higher-order derivative terms, but are not pursued here. The associated FPE describes the time evolution of the PDF $p(\vec{\mathbf{x}}, t)$ of solutions of the SDE.

$$\frac{\partial p(\vec{\mathbf{x}}, t)}{\partial t} = -\sum_{i=1}^{d} \frac{\partial}{\partial x_i}[\mu_i(\vec{\mathbf{x}}, t)p(\vec{\mathbf{x}}, t)] + \sum_{i=1}^{d}\sum_{j=1}^{d} \frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\vec{\mathbf{x}}, t)p(\vec{\mathbf{x}}, t)\right] \tag{6}$$

Instead of numerically solving the FPE, in this work, we use a Monte Carlo (MC) method to estimate $p(\vec{\mathbf{x}}, t)$ by simulating a large number of instances of an SDE. Explicit numerical solution of an SDE involves choosing an initial position $\vec{\mathbf{x}}_0$ and iteratively updating it to get the value at time t, $\vec{\mathbf{x}}_t$. For the MC simulation, we start with a set of initial positions $\{\vec{\mathbf{x}}_0\}$ sampled from the desired initial distribution and numerically solve each of them independently. We can then use the set of samples at time t, $\{\vec{\mathbf{x}}_t\}$, to compute the desired statistics. We can formally write this iteration step as follows:

$$\vec{\mathbf{x}}_{t+\Delta t} = \mathcal{M}\left(\vec{\mathbf{x}}_t, \vec{\boldsymbol{\mu}}(\vec{\mathbf{x}}_t, t), \boldsymbol{\sigma}(\vec{\mathbf{x}}_t, t), \Delta\vec{\mathbf{W}}_t, \Delta t\right) \tag{7}$$

Here, $\mathcal{M}$ denotes an arbitrary update scheme. There are several methods [38] by which we can create this update. Throughout this work, we will be working with autonomous SDEs ($\vec{\boldsymbol{\mu}}$ and $\boldsymbol{\sigma}$ do not explicitly depend on time) with diagonal noise ($\boldsymbol{\sigma}$ is a diagonal matrix). Therefore, we will use a simple update scheme called the Milstein scheme [39]:

$$\vec{\mathbf{x}}_{t+\Delta t} = \vec{\mathbf{x}}_t + \vec{\boldsymbol{\mu}}(\vec{\mathbf{x}}_t)\Delta t + \boldsymbol{\sigma}(\vec{\mathbf{x}}_t).\Delta\vec{\mathbf{W}}_t + \frac{1}{2}\boldsymbol{\sigma}(\vec{\mathbf{x}}_t).\boldsymbol{\sigma}'(\vec{\mathbf{x}}_t).\left(\left(\Delta\vec{\mathbf{W}}_t\right)^2 - \Delta t\right) + \mathcal{O}\left(\Delta t^{1.5}\right) \tag{8}$$

Here, $\Delta\vec{\mathbf{W}}_t$ is a random sample from $\mathcal{N}^d(0, \Delta t)$. Note that the Milstein scheme is only accurate up to $\mathcal{O}(\Delta t)$, and this error can accumulate through the course of the simulation. Therefore, to control the error in the numerical update scheme, we will adaptively choose the time step Δt for updates by setting a local error tolerance. Local error at time t is defined as the deviation between a single step update made with time step Δt and a two-step update made with the time step $\Delta t/2$ each.

$$err(\vec{\mathbf{x}}_t, \Delta t) := \frac{\left\| \left(\mathcal{M}\left(\vec{\mathbf{x}}_t, \Delta\vec{\mathbf{W}}_1, \frac{\Delta t}{2}\right) + \mathcal{M}\left(\vec{\mathbf{x}}_{t+\frac{\Delta t}{2}}, \Delta\vec{\mathbf{W}}_2, \frac{\Delta t}{2}\right)\right) - \mathcal{M}\left(\vec{\mathbf{x}}_t, \Delta\vec{\mathbf{W}}_t, \Delta t\right) \right\|}{\sqrt{d}} \tag{9}$$

Here, we omitted the dependence of $\mathcal{M}$ on $\vec{\boldsymbol{\mu}}$ and $\boldsymbol{\sigma}$ for brevity. $\Delta\vec{\mathbf{W}}_1$ and $\Delta\vec{\mathbf{W}}_2$ are chosen in such a way that $\Delta\vec{\mathbf{W}}_1 + \Delta\vec{\mathbf{W}}_2 = \Delta\vec{\mathbf{W}}_t$. To satisfy the local error tolerance Tol, we need to choose Δt, such that $err(\vec{\mathbf{x}}_t, \Delta t) < Tol$. The exact prescription on how this choice is made can be found here [40].

Computing Γ requires computing the derivative of $P(\vec{\mathbf{x}}_t, t)$ of the numerical solutions. Since derivatives are sensitive to numerical noise, we need accurate estimates of the probability distribution. This is achieved by numerically solving a large number of SDEs (we use at least 2×10^7 samples in this work). This is an impractically large number for most computers. Numerically finding 2×10^7 solutions, with each requiring around 10,000 time steps (depending on the required accuracy) will require at least 1600 GB of memory, assuming 64-bit floating point values. Additionally, assuming a fast 2 ms per solution, the entire simulation will take around 11 h if carried out serially. To solve these problems, we use GPU computing. With GPU computing [41], we can perform updates on a set of values $\{\vec{\mathbf{x}}_t\}$, simultaneously using Equation (7) to get $\{\vec{\mathbf{x}}_{t+\Delta t}\}$. $\{\vec{\mathbf{x}}_t\}$ can then be removed from memory after computing the desired statistics, making it memory efficient. As for the runtime, a GPU-based parallel implementation [42] in Python takes around 4 min to simulate 2×10^7 samples for 10,000 time steps on a consumer laptop equipped with Nvidia RTX 2080 GPU. See Appendix A for detailed scaling relations of simulation runtime.

2.2. Estimating Γ

The form of Γ in Equation (4) makes it unsuitable for numerical computation due to the presence of $p(\mathbf{x}, t)$ term in the denominator, which can become zero. We therefore rewrite the equation using the redefinition $q(\mathbf{x}, t) := \sqrt{p(\mathbf{x}, t)}$.

$$\Gamma^2(t) = 4 \int d\mathbf{x} \left[\frac{\partial q(\mathbf{x}, t)}{\partial t}\right]^2 \tag{10}$$

To calculate Γ, we first estimate the PDF using histograms. (It would be more accurate to use kernel density estimators [43], but that is more computationally expensive). The derivative in Equation (10) is approximated as a finite difference and the integral as a Riemann sum. The specific methods and the error estimates are provided in Appendix B. After computing Γ, information length can be computed from the numerical integration of Equation (3).

A major source of error in the estimation of Γ is due to the compact support and overlap of numerically estimated probability densities. (This will also be a source of error with numerical FPE solvers). During the simulation, time steps are chosen adaptively by setting a tolerance for the local numerical error. However, in some regimes, this results in the system evolving too fast, so that there is little or no overlap between the probability densities at adjacent time steps (Figure 1). Theoretically, these densities (Figure 1) have the support of the entire real line, and the only error will be caused by approximating the derivative and the integral. However, in practice, we are running the MC simulation with a finite sample size, and computers have finite numerical precision; as such, the estimated densities have compact support. Figure 2 shows how the amount of overlap between the densities affects the error in Γ estimate. There are two main sources of sub-optimal overlap: changes in mean and changes in standard deviation.

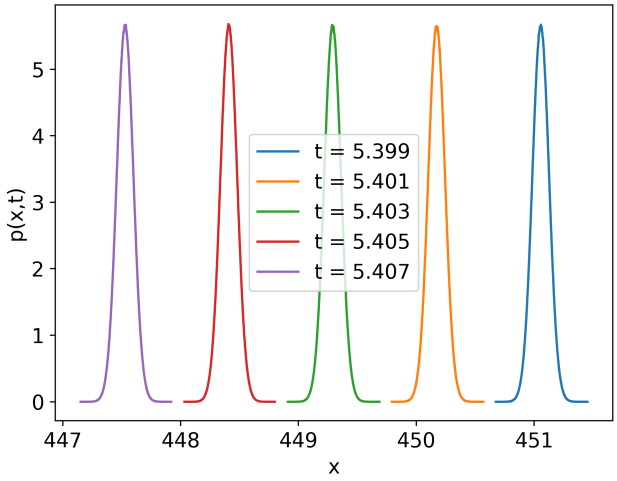

Figure 1. Probability density at adjacent time steps for the SDE $dx_t = -x_t dt + 0.1 dW_t$ with initial normal distribution $\mathcal{N}(10^5, 10^{-10})$. Time steps were chosen adaptively to limit local error to 5×10^{-4}. The specific time interval was chosen to showcase the lack of overlap between PDFs.

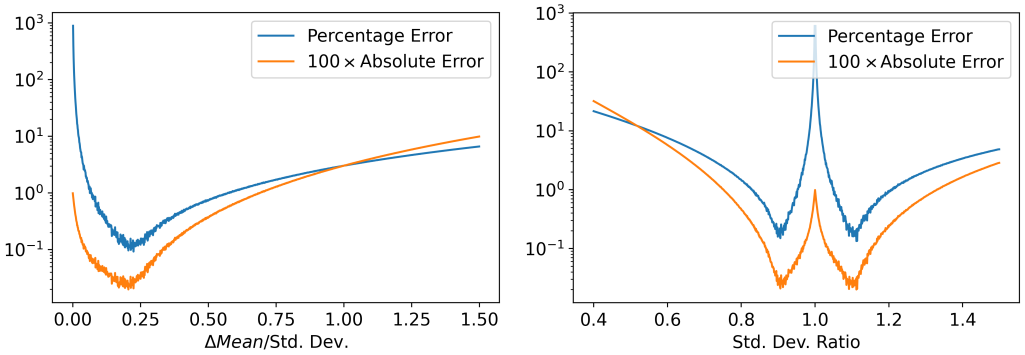

Figure 2. (**left**) Error in Γ calculation for two Gaussians with same standard deviation (Std. Dev. = 1) but with different means. The error estimate of Γ is lowest when ΔMean $\approx$ 0.2 Std. Dev. (**right**) Error in Γ calculation for two Gaussians with same Mean (Mean = 0) but with different standard deviation. The error Γ estimate is lowest when ratio of standard deviation is approximately 0.9 or 1.1 ($\approx 0.9^{-1}$). Discretized version of Equation (10) was used for the estimate and PDF was approximated by a histogram with 703 bins. We considered 2×10^7 samples for each distribution. dt is chosen to be 1. Estimates were repeated 40 times and mean of the error was taken.

Consider Δt chosen adaptively to satisfy the local error tolerance. Now assume for some $\widetilde{\Delta t}$, we get optimal overlap. If $\widetilde{\Delta t} > \Delta t$, we can wait for a few steps before estimating the Γ. However, if $\widetilde{\Delta t} < \Delta t$, we can choose a $\widetilde{\Delta t}$ as a temporary time step and generate a

collection of points $\{\vec{\mathbf{x}}_{t+\widetilde{\Delta t}}\}$ from $\{\vec{\mathbf{x}}_t\}$ using Equation (7), which results in optimal overlap between densities. The local error will still be smaller than the tolerance since $\Delta t < \widetilde{\Delta t}$. In order to derive the value of $\widetilde{\Delta t}$, we use the Milstein update scheme and restrict ourselves to the one-dimensional case for simplicity. First, we look at the change in mean value.

$$x_{t+\Delta t} = x_t + \mu(x_t)\Delta t + \sigma(x_t)(x_t)\Delta W_t + \frac{1}{2}\sigma(x_t)\sigma'(x_t)\left(\left(\Delta W_t^2\right) - \Delta t\right) + \mathcal{O}\left(\Delta t^{1.5}\right) \quad (11)$$

Taking the expectation value of both sides, we get:

$$\mathbb{E}[x_{t+\Delta t}] = \mathbb{E}[x_t] + \mathbb{E}[\mu(x_t)]\Delta t + \mathcal{O}\left(\Delta t^{1.5}\right) \quad (12)$$

Here, we use the fact that $\mathbb{E}[\Delta W_t] = 0$ and $\mathbb{E}\left[(\Delta W_t)^2\right] = \Delta t$. Note that x_t and ΔW_t are independent random variables. For X and Y, the independent random variables are $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$.

In order to achieve optimal overlap (Figure 2 (left)), we need $\mathbb{E}[x_{t+\Delta t}] - \mathbb{E}[x_t] = 0.2\,\mathrm{Std}[x_t]$.

$$\widetilde{\Delta t}_{mean} = \frac{0.2\,\mathrm{Std}[x_t]}{\mathbb{E}[\mu(x_t)]} \quad (13)$$

Now to derive the effect on change in standard deviation on $\widetilde{\Delta t}$, a similar calculation for $\mathrm{Var}[x_{t+\Delta t}]$ can be performed, which yields:

$$\mathrm{Var}[x_{t+\Delta t}] = \mathrm{Var}[x_t] + \left(2\,\mathrm{Cov}[\mu(x_t), x_t] + \mathbb{E}\left[\sigma^2(x_t)\right]\right)\Delta t + \mathcal{O}\left(\Delta t^{1.5}\right) \quad (14)$$

Now, in order to achieve optimal overlap, we need $\mathrm{Var}[x_{t+\Delta t}]/\,\mathrm{Var}[x_t] = 0.9^{\pm 2}$ (Figure 2 (right)). 0.9^{+2} when $\mathrm{Var}[x_{t+\Delta t}] < \mathrm{Var}[x_t]$ and 0.9^{-2} when $\mathrm{Var}[x_{t+\Delta t}] > \mathrm{Var}[x_t]$.

$$\widetilde{\Delta t}_{std} = \mathrm{Var}[x_t]\left|\frac{0.9^2 - 1}{2\,\mathrm{Cov}[\mu(x_t), x_t] + \mathbb{E}[\sigma^2(x_t)]}\right| \quad (15)$$

Note that we have only considered the first and second moments here. It is potentially possible to improve the accuracy of the Γ estimate by considering higher moments. However, this improvement will be marginal, since overlap between two distributions is most affected by its first and second moments.

After calculating both the time steps $\widetilde{\Delta t}_{mean}$ and $\widetilde{\Delta t}_{std}$, we take the minimum of the two to perform the update on $\{x_t\}$ to get $\{x_{t+\widetilde{\Delta t}}\}$, where $\widetilde{\Delta t} = \min\left(\widetilde{\Delta t}_{mean}, \widetilde{\Delta t}_{std}\right)$. After the estimation, the Γ $\{x_{t+\widetilde{\Delta t}}\}$ is discarded and we continue the simulation with $\{x_{t+\Delta t}\}$. This prevents any significant slowing in the simulation if $\widetilde{\Delta t} \ll \Delta t$. Note that calculating $\widetilde{\Delta t}_{mean}$ and $\widetilde{\Delta t}_{std}$ using the entire set of points $\{x_t\}$ will be computationally expensive. Only a small subset is used to perform this calculation.

After Γ is estimated, we can calculate the information length by approximating the integral as a Riemann sum.

Looking at Figure 3, it can be seen than the percentage error in Γ blows up towards the end of the simulation. This is when the system approaches a stationary state and the probability density stops evolving. The exact Γ reaches zero, whereas the numerical estimate will have a small nonzero error (Figure 4). Even though the percentage error in Γ becomes large, the absolute error remains small (Figure 2 (left)) and will have minimal contribution to the error in Information Length calculation. However, when the initial distribution is 'closer' to the stationary distribution (small x_0 values in Figure 3), the absolute value of Information Length will be small, and the error in Γ estimate will have a more significant contribution to the Information Length calculation.

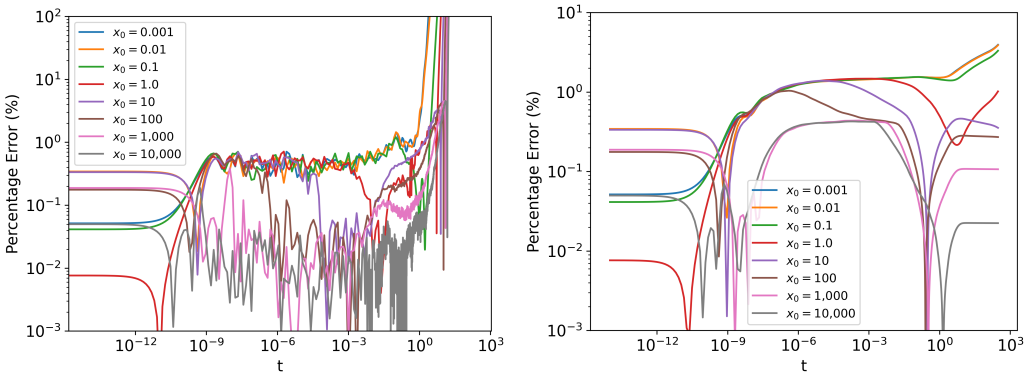

Figure 3. Error in estimated (**left**) Γ and (**right**) $\mathcal{L}$ by simulating 2×10^7 parallel instances of the Equation $dx_t = -1.0x_t dt + 0.1dW_t$ with initial values sampled from the normal distribution $\mathcal{N}(x_0, 10^{-10})$ and computing the PDFs using histograms with 703 bins. The local error tolerance was 5×10^{-4}. Note that an initial step size $\Delta t = 10^{-14}$ was chosen to produce more frequent estimates in the initial part of the simulation. See Appendix D for exact solution.

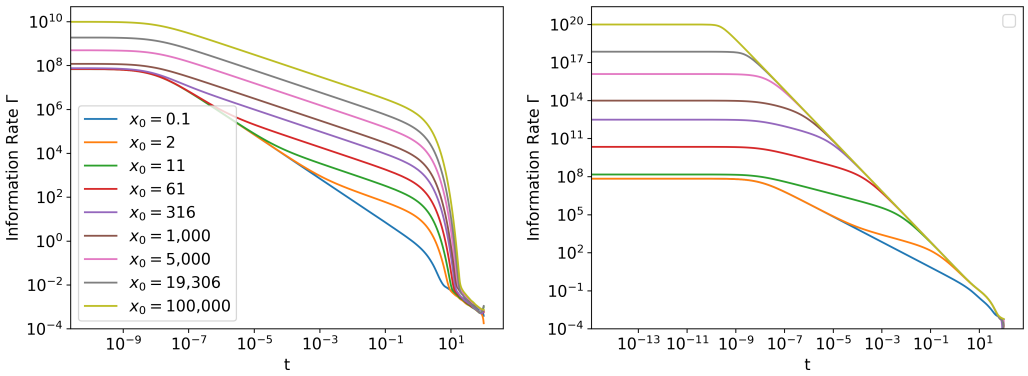

Figure 4. The Γ of (**left**) Linear SDE and (**right**) Cubic SDE for different initial conditions $\mathcal{N}(x_0, 10^{-10})$, as described in Section 3. Note that towards the end of the range of t, the values will be dominated by errors, as shown in Figure 3. Therefore instead of dropping to zero, they will have a finite nonzero value. For exact solution of Linear SDE, see Appendix D.

It is to be noted that measuring Γ of multi-dimensional problems is a significant computational challenge which requires further investigation. MC SDE simulation is better at handling such problems compared with FPE solvers due to its linear scaling with number of dimensions. However, this comes at the price of accuracy when estimating PDFs of higher dimensional problems. It is still possible to accurately study Γ from marginal PDFs of multi-dimensional problems using MC SDE simulation.

A python implementation of SDE Simulation, along with the Γ measurement used in this work, can be found here [42].

3. Linear vs. Cubic Statistics

In this section, we will use methods developed in the previous section and study nonlinear damping of the Information Geometry of a stochastic process. We will compare the Ornstein–Uhlenbeck process [44], a model for prototypical linear driven dissipative process, defined by linear SDE:

$$dx_t = -\theta x_t dt + \sigma dW_t \tag{16}$$

with a cubic SDE defined by the equation:

$$dx_t = -\theta x_t^3 dt + \sigma dW_t \tag{17}$$

The cubic damping term has been previously used to model CO_2 emissions [45], phase transitions [24], and self-organized shear flows [46].

Throughout this section, unless otherwise stated, we restrict ourselves to the values $\theta = 1.0$ and $\sigma = 0.1$. The initial distribution is always $\mathcal{N}(x_0, 10^{-10})$, and we run the SDE simulation from $t = 0$ to $t = 100$. The data in this section were generated using 2×10^7 samples, and PDF was approximated by a histogram with 703 bins, which was chosen using an empirical formula $2.59\sqrt[3]{n}$. The time steps were adaptively chosen by setting local error tolerance at 5×10^{-4}.

Two of the simplest statistics that can be measured are the mean and the standard deviation of the distribution (Figures 5 and 6). The trends in mean can be readily seen by taking expectation value on both sides of Equations (16) and (17). For the linear SDE, we have $d\mathbb{E}[x_t] = -\theta\mathbb{E}[x_t]dt$, which can be solved to get $\mathbb{E}[x_t] = x_0 e^{-t}$. For the Cubic SDE, we can follow similar steps:

$$d\mathbb{E}[x_t] = -\theta\mathbb{E}\left[x_t^3\right]dt \tag{18}$$

$$d\mathbb{E}[x_t] \approx -\theta\mathbb{E}[x_t]^3 dt \tag{19}$$

$$\implies \mathbb{E}[x_t] \approx \frac{1}{\sqrt{\frac{1}{x_0^2} + 2\theta t}} \tag{20}$$

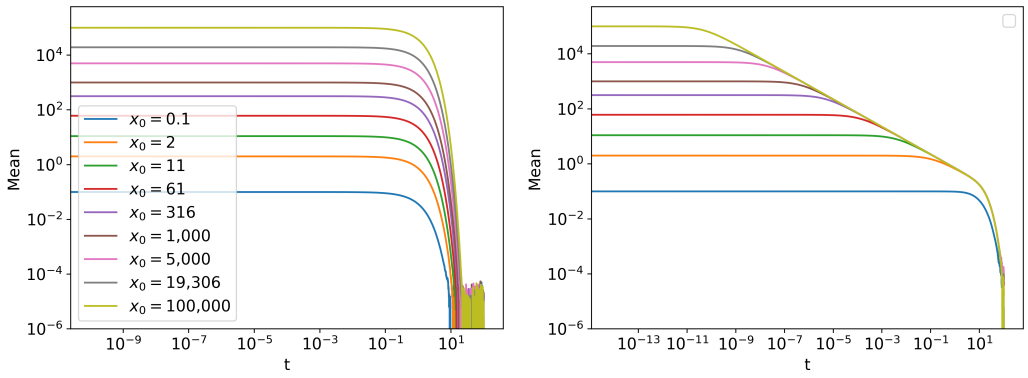

Figure 5. The mean of the distribution for different initial conditions $\mathcal{N}(x_0, 10^{-10})$ of (**left**) linear SDE and (**right**) cubic SDE.

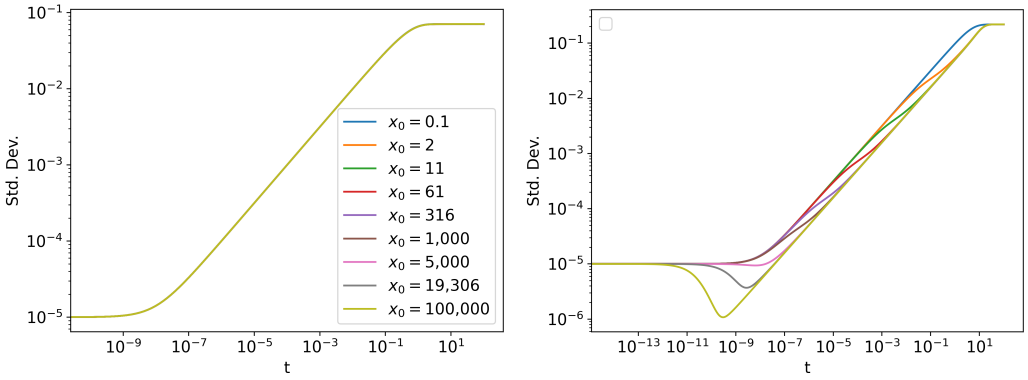

Figure 6. The standard deviation of the distribution for different initial conditions $\mathcal{N}(x_0, 10^{-10})$ of (**left**) linear SDE and (**right**) cubic SDE. Note that for the linear SDE, all the lines overlap.

The approximate solution of the mean value of Cubic SDE is only valid when the standard deviation of the distribution is much smaller than the mean (Appendix C). Note that in Equation (20), when $1/x_0^2 \ll \theta t$, $\mathbb{E}[x_t] \approx (2\theta t)^{-0.5}$, making the trajectory independent of x_0. This can be seen in Figure 5 (right), when all the lines merge into one. However, around $t = 5$, the approximation fails, since the value of the standard deviation (Figure 6 (right))

and the mean (Figure 5 (right)) becomes comparable. Therefore, the curves fail to follow the same trajectory $\mathbb{E}[x_t] \approx (2\theta t)^{-0.5}$ for $t > 5$.

The trends in standard deviation of the Linear SDE are readily explained by Equation (A14), which does not depend on initial mean position x_0, but only on the initial standard deviation, the drift coefficient, and the diffusion coefficient. As we write this paper, no exact analytic solution exists for the Cubic SDE. However, an approximate analytic treatment can be found here [47].

We define the asymptotic Information Length ($\mathcal{L}_\infty$) to be the Information Length it took for the system to reach the stationary state of its PDF.

$$\mathcal{L}_\infty := \mathcal{L}(t \to \infty) \tag{21}$$

Analytically, a SDE reaches its stationary state as $t \to \infty$. However, numerically, we see that the probability density stops evolving after a finite time. We can see this from Figure 7, as $\mathcal{L}$ becomes a constant. $\mathcal{L}$ will still continue to increase slightly due to numerical error, but this contribution will be negligible, as shown in Figure 3. $\mathcal{L}_\infty$ measures how many statistically distinguishable states the system passes through to reach its stationary state. From Figure 6, it is evident that, compared to the linear SDE, the PDF of the cubic SDE undergoes a lot more change before reaching its stationary state, for larger values of x_0. For small values of x_0, the trend in standard deviation is similar between linear and cubic SDEs, since the initial evolution of the cubic process is Gaussian. This is confirmed in Figure 7, which shows that for large values of x_0, $\mathcal{L}_\infty$ is significantly larger for the cubic SDE for same values of x_0 and, for small x_0 values, the $\mathcal{L}_\infty$ values are comparable.

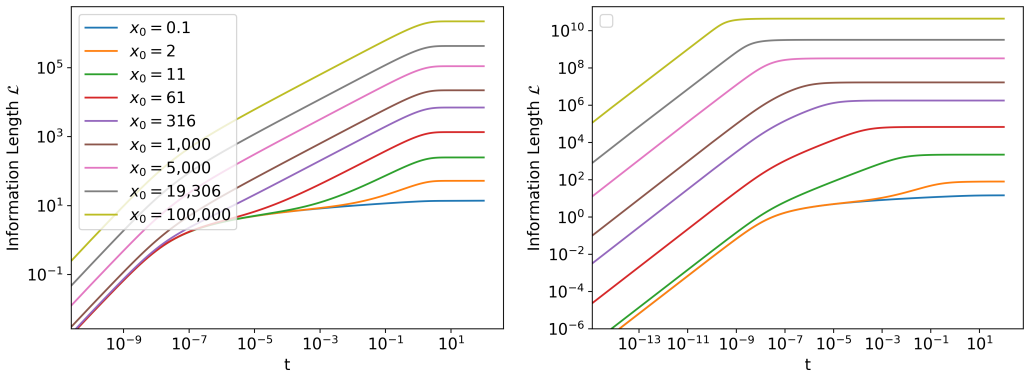

Figure 7. The Information Length of (**left**) linear SDE and (**right**) cubic SDE for different initial conditions $\mathcal{N}(x_0, 10^{-10})$. For exact solution of linear SDE, see Appendix D.

Information Length Scaling

In Figure 7, we have already seen that $\mathcal{L}_\infty$ depends on x_0 and has different behavior for linear and cubic SDEs. In [47], by numerically solving the FPE (also analytically for the linear SDE), it was shown that for large values of x_0, $\mathcal{L}_\infty$ shows different scaling behavior for linear and cubic SDEs. For the linear SDE, $\mathcal{L}_\infty \sim x_0$. For the cubic SDE, $\mathcal{L}_\infty \sim x_0^m$, where $1.5 < m < 1.9$. We reproduce this result in Figure 8. Note that, since $\mathcal{L}_\infty$ is a dimensionless quantity, it is not possible to derive these values theoretically using dimensional analysis. In the absence of general analytic tools to study this property, numerical methods are indispensable.

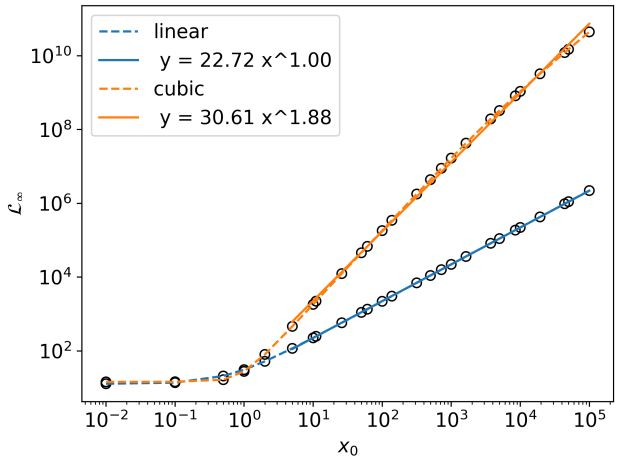

Figure 8. Scaling behavior of $\mathcal{L}_\infty$ with respect to x_0 for linear and cubic SDE with $\theta = 1.0$ and $\sigma = 0.1$.

4. Unequal Time Joint PDF: Bimodality for Cubic Force

A clear advantage of MC SDE Simulation over solving FPE is the ability to compute the unequal time joint PDFs $P(x(t_1), x(t_2))$. Unequal time joint PDFs have been previously used for causality and to establish causal relations. In this section, we showcase the ability of MC SDE simulation to estimate $P(x(t_1), x(t_2))$ and study its properties.

4.1. Unequal Time PDFs in the Stationary State

For the linear SDE $P(x(t_1), x(t_2))$ is always a Gaussian, but with a covariance matrix that depends on t_1 and t_2. However, for the cubic SDE, we see the emergence of a bimodal distribution (Figure 9 (right)) depending on t_1 and t_2 values. This behavior prevails even after the system has reached its stationary state, but now only depends on the difference $\Delta t := t_2 - t_1$. The bimodality indicates a finite memory induced by the non-linearity. In order to further study this behavior, we first ensure the system has reached a stationary state by simulating from $t = 0$ to $t = 1500$ with a sample size of 8×10^7 for the cubic process. After setting $t_1 = 1500$, t_2 values are chosen from the data generated by further simulating the system for 300 time units. The PDF was approximated by a histogram with 60×60 bins. The time steps were adaptively chosen by setting the local error tolerance as 0.01σ.

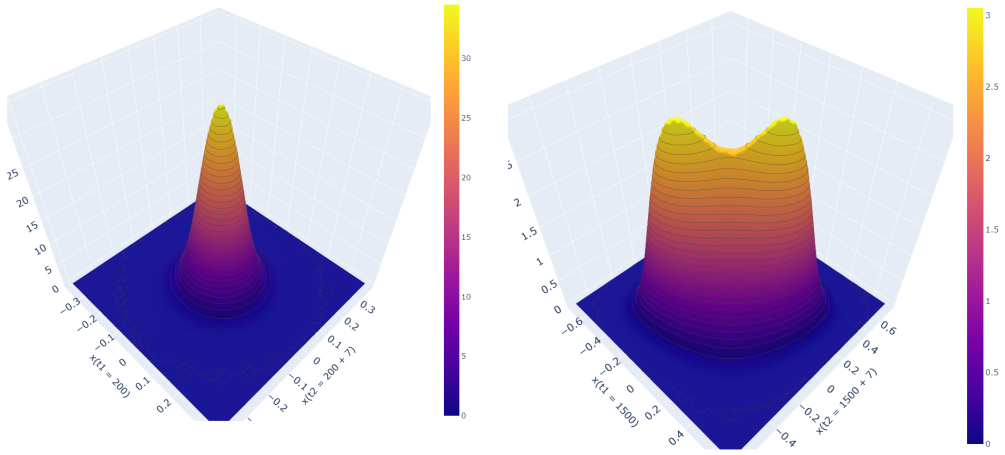

Figure 9. $P(x(t_1), x(t_2))$ for (**left**) linear SDE with $t_1 = 200$ and $t_2 = 207$ and (**right**) cubic SDE with $t_1 = 1500$ and $t_2 = 1507$. For both the SDEs, $\gamma = 1.0$ and $\sigma = 0.1$. Both SDEs have reached their stationary states.

Note that, because of the symmetry of diffusion term and the drift term, the bimodality of $P(x(t_1), x(t_2))$ for cubic SDE is always symmetric with respect to the line $x(t_1) = x(t_2)$. Therefore to make the numerical study of the bimodality easier, we restrict our attention to the diagonal of the joint probability density $P(x(t_1) = x(t_2))$, which is then normalized to integrate to one.

To quantify the bimodality of $P(x(t_1) = x(t_2))$, we use the following fitting function to perform a nonlinear fit and estimate the parameters.

$$P(x(t_1) = x(t_2)) = a\left[\exp\left(-\left(\frac{x}{c}\right)^4\right) + b\exp\left(-\left(\frac{x-e}{d}\right)^2\right) + b\exp\left(-\left(\frac{x+e}{d}\right)^2\right)\right] \tag{22}$$

To motivate this fitting function, note that for $\Delta t = 0$, we get a purely quartic exponential, since it is nothing but the stationary state $P_s(x)$ of the cubic SDE. For $\Delta t \to \infty$, the correlation between points reaches zero, $P(x(t_1), x(t_2)) \sim P_s(x_1(t_1))P_s(x_2(t_2))$, the product of two independent stationary distributions. The quadratic exponential terms are motivated by Gaussian distribution of the noise, and are found to describe the data accurately. There are two quadratic exponential terms, since the SDE is symmetric about the point $x = 0$ and the bimodal peaks occur symmetrically on opposite sides.

In the fitting function, Parameter a is a measure of the overall height of the density curve. Parameter b represents the ratio of contribution from the quadratic exponential to the quartic exponential, and denotes the degree of bimodality. Parameter e is the location of the peaks. Parameter c is a measure of the overall spread of the density function, while parameter d is a measure of the spread of bimodal peaks. Note that a nonzero value for parameter e and b denotes bimodality in the distribution.

From Figure 10, we can see that the joint PDF becomes bimodal for a range of Δt values, since parameters b and e have nonzero values. The standard deviation of the quadratic term denoted by parameter d is almost a constant for a fixed σ, whereas the location of the bimodal peak denoted by parameter e changes slightly, but never becomes zero. That means that while transitioning from a bimodal to a unimodal distribution, the bimodal peaks do not continuously merge into one another, but slowly become less prominent and eventually disappear, as inferred from the value of parameter b. The artifacts towards the end of the curves of parameters b and e are due to the fact that it is not possible to consistently fit parameters b and e when there is negligible contribution from the quadratic exponential term as $b \to 0$.

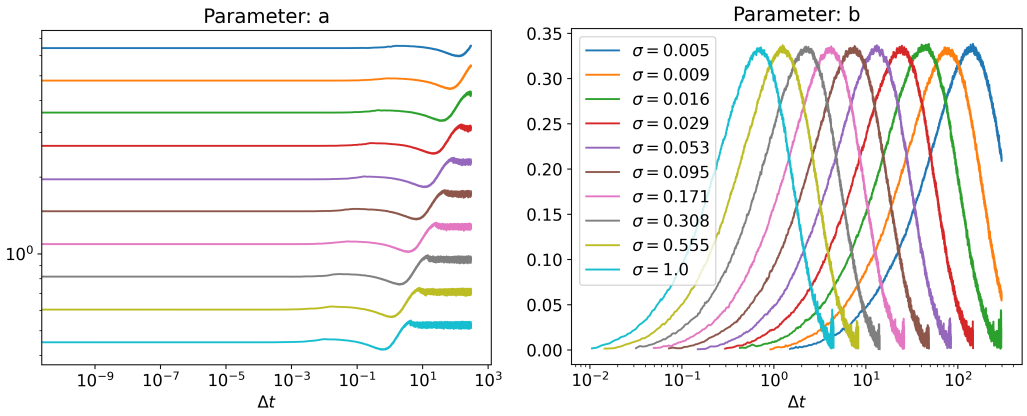

Figure 10. *Cont.*

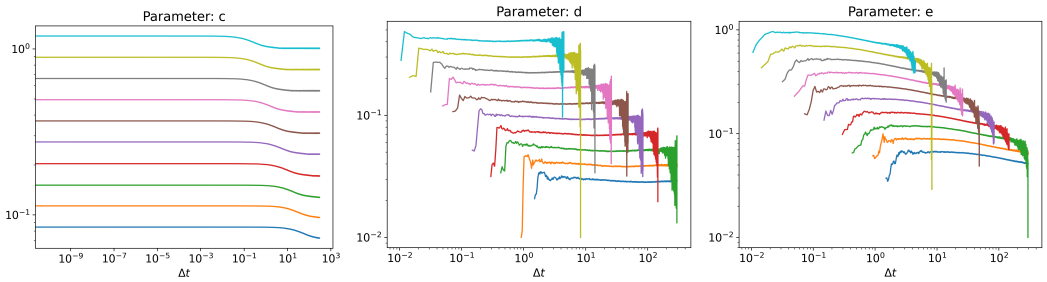

Figure 10. Panels show the values of a, b, c, d and e parameters of Equation (22), obtained through nonlinear function fitting at each time point t_2, expressed as a function of $\Delta t := t_2 - t_1$ for different noise levels σ. To ensure the density function $p(x,t)$ for the cubic process reached a stationary state, we chose $t_1 = 1500$. Note that for parameters b, d and e, the domain of the plots is restricted to the region where parameter b is nonzero. Noise starts dominating outside this region, since there is negligible contribution from the quadratic terms, and nonlinear fit cannot find a consistent unique value for these parameters.

The values of parameter a and c in Figure 10 can be understood by first considering the $\Delta t \to 0$ and $\Delta t \to \infty$ limits. For $\Delta t \to 0$, we have $P(x(t_1), x(t_2)) = \delta(x(t_1) - x(t_2))P_s(x)$, where $P_s(x)$ is the stationary state. The diagonal of joint PDF then becomes $P(x(t_1) = x(t_2)) = P_s(x) := a_s \exp\left(-(x/c_s)^4\right)$. Here, after integrating x, we can find, $a_s \approx 0.55/c_s$ for all values of σ, which can be numerically verified. When $\Delta t \to \infty$, since there is no correlation, $P(x(t_1), x(t_2)) = P_s(x(t_1))P_s(x(t_2))$. The normalized diagonal then becomes $P(x(t_1) = x(t_2)) \sim \exp\left(-2(x/c_s)^4\right)$. The normalization factor can be derived by integrating out x. Therefore, when $\Delta t \to \infty$, we have $a = 2^{1/4} a_s$ and $c = c_s/2^{1/4}$, which agrees with numerical results. For intermediate Δt values, there will be correlation, and the behavior cannot be easily explained.

The trends in Figure 10 for different σ values can be explained by looking at scaling relations. For the cubic equation, we have $dx_t = -\theta x_t^3 dt + \sigma dW_t$. Looking at the individual terms, we can infer the dimensions, $\theta \sim x^{-2} t^{-1}$ and $\sigma \sim x^1 t^{-1/2}$. Therefore, we expect the following scaling relations:

$$t \sim \frac{1}{\theta^{1/2}\sigma} \tag{23}$$

$$x \sim \theta^{-1/2}\sigma^{1/2} \tag{24}$$

These relations are numerically verified in Figure 11 for a fixed value of θ, for parameters a, c, d, e, and the relationship between noise and Δt, corresponding to peak value of parameter b. Note that $a \sim \sigma^{0.5}$ in Figure 11, because parameter a is a normalization constant. $\int ap(x)dx = 1 \implies a \sim x^{-1} \sim \theta^{1/2}\sigma^{-1/2}$. The peak value of parameter b seems to be a constant with the value of $1/3$. It cannot be derived from simple scaling arguments alone; further investigation is required to understand its origin.

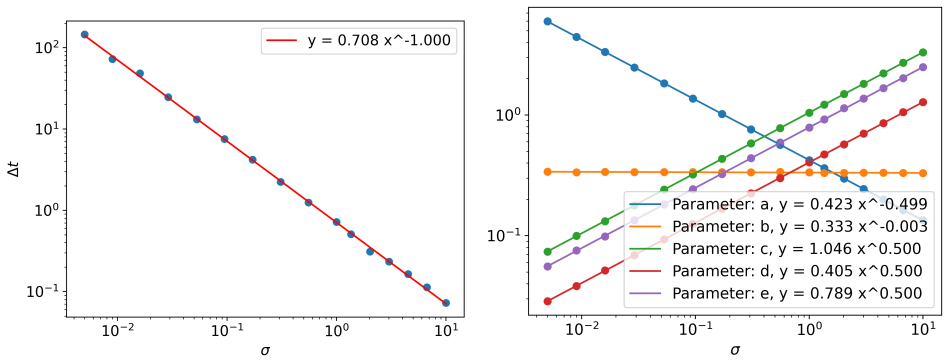

Figure 11. (**left**) Scaling behavior of peak poisition of parameter b with respect to noise level σ and (**right**) Scaling behavior of parameter values corresponding to the peak position of parameter b.

4.2. Evolution of Bimodality in the Non-Stationary State

In this section, we will look at how the bimodality in the joint PDF evolves qualitatively before $p(x,t)$ has reached a stationary state. Since bimodality occurs only for the cubic process, the following results are for the SDE $dx_t = -1.0x_t^3dt + 0.1dW_t$. To this end, we follow the same prescription from Section 4, but with a slightly modified version of the fitting function (Equation (22)).

$$P(x(t_1) = x(t_2)) = a\exp\left(-\left(\frac{x}{c}\right)^4\right) + b\exp\left(-\left(\frac{x-e}{d}\right)^2\right) + b\exp\left(-\left(\frac{x+e}{d}\right)^2\right) \quad (25)$$

This modification is made since for some t_1 and t_2 values, there is no contribution from the quartic exponential term, unlike in the case of stationary state, where there is always a quartic exponential contribution to the distribution.

The nonstationary phase exhibits (Figure 12) rich behavior, which asymptotically transitions to the stationary state behavior as t_1 becomes large. For small t_1 values, Parameter $a \ll$ Parameter b, since $P(x(t_1), x(t_2))$ has very little contribution from the quartic exponential term. This is because the initial distribution is a Gaussian distribution, and it is slowly evolving towards the quartic exponential distribution in the stationary state. For larger t_1 values, Parameter a starts dominating, since $P(x(t_1), x(t_2))$ has predominant contribution from quartic exponential term, as expected of the system reaching its quartic exponential stationary state. For some intermediate t_1 values, depending on Δt values, we see an interesting behavior where $P(x(t_1), x(t_2))$ becomes purely a quartic exponential (Parameter b = 0) twice before becoming a mixture of quadratic and quartic exponential terms. Further investigation into this behavior is not undertaken at this time, and only serves to demonstrate the potential capabilities of GPU-accelerated MC SDE simulation for future work.

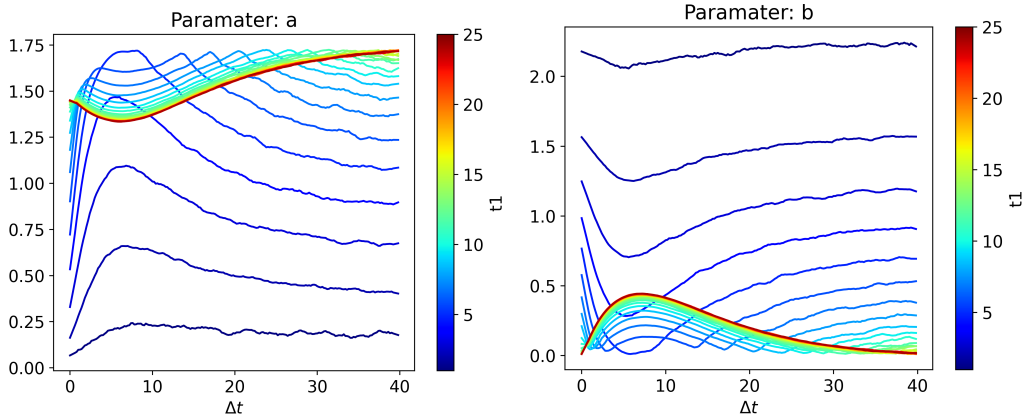

Figure 12. Behavior of (**left**) Parameter a and (**right**) Parameter b for different t_1 and $t_2 = t_1 + \Delta t$ values.

5. Discussion

In this work, we developed a method for fast and accurate study of the Information Geometry of SDEs using Monte Carlo simulation. We identified the computational challenges and overcame them by using GPU computing. Specifically, the major limitation with MC SDE simulation in the estimation of Γ was the sub-optimal overlap of PDFs at subsequent time points. We solved this problem by developing an interpolation method to compute PDFs with optimal overlap.

As an application of the new method, we compared the Information Geometry of SDEs with a linear and a cubic damping force. We were able to reproduce the analytic results for the linear SDE and the previous numerical results [34] for cubic SDE obtained using FPE solvers. This was particularly true of large values of x_0, $\mathcal{L}_\infty \sim x_0$ for the linear case and $\mathcal{L}_\infty \sim x_0^m$ for the cubic case, where $m = 1.88$ when $\theta = 1.0$ and $\sigma = 0.1$. We further

showcased the advantage of MC SDE simulation over FPE solver by computing the joint PDF $P(x(t_1), x(t_2))$. Unlike the linear SDE, the cubic SDE led to an interesting bimodal PDF $P(x(t_1), x(t_2))$, which was observed even after reaching a stationary state. After reaching the stationary state $P(x(t_1), x(t_2))$ only depends on $\Delta t = t_2 - t_1$. In the stationary state, we further studied the bimodality by quantifying it and looking at the power-law scaling relations with respect to noise levels σ and provided theoretical scaling arguments. The maximum value of the ratio of quadratic to quartic contribution (Parameter b) is found to be a constant $1/3$ irrespective of the noise levels, which requires further study. Finally, we qualitatively looked at $P(x(t_1), x(t_2))$ in the nonstationary state. The MC SDE simulation can be an important tool for further studying this behavior.

It is important to note that the methods that we developed here for one stochastic variable are general, and can be extended for more than one variable, as well as for investigating different metrics or thermodynamic quantities. These will be addressed in future work. Furthermore, it will be of interest to investigate fast implementations of Kernel Density Estimators [48,49] which will improve the accuracy of joint PDF estimates. We note that in numerical experiments, it was seen that compared with histograms, using Kernel Density Estimators to estimate PDFs provided 2–5 times reduction in error with identical PDFs, albeit with a performance trade-off.

Author Contributions: Conceptualization, A.A.T. and E.-j.K.; Formal analysis, A.A.T.; Investigation, A.A.T. and E.-j.K.; Methodology, A.A.T.; Project administration, E.-j.K.; Resources, A.A.T.; Software, A.A.T.; Supervision, E.-j.K.; Validation, A.A.T. and E.-j.K.; Visualization, A.A.T.; Writing—original draft, A.A.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, A.A.T., upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SDE	Stochastic Differential Equation.
FPE	Fokker–Planck Equation.
PDF	Probability Density Function.
MC	Monte Carlo.
GPU	Graphics Processing Unit.

Appendix A. Runtime Scaling

We use the linear O-U process $dx_t = -\theta x_t dt + \sigma dW_t$ to study the runtime scaling relations of MC SDE simulation. Simulations start with n samples from an initial distribution $\mathcal{N}(x_0, 10^{-10})$. The simulation were run for 100 time units on a workstation with AMD EPYC 7451 24-Core CPU and Nvidia Titan RTX GPU, using the implementation here [42]. Milstein scheme was used with adaptive time steps with local error tolerance Tol. Unless otherwise stated, the parameters will have the values: $x_0 = 10$, $\sigma = 0.1$, $\theta = 0.1$, $Tol = 5 \times 10^{-4}$ and $n = 2 \times 10^6$. Note that computation only involves the MC SDE simulation. Calculating the statistics and probability density will require additional computation. This might increase the overall computational time, but the value of the power in the scaling relationship will remain the same.

For MC SDE simulation, as seen in Figure A1, initially the runtime weakly depends on n since computations are being done in parallel. But when n is significantly larger than

the number of parallel processors, the computations need to be done in batches and the runtime scales linearly.

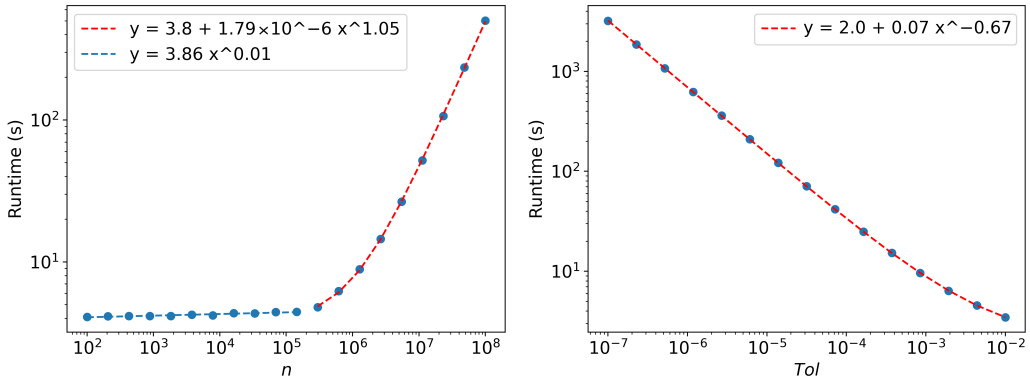

Figure A1. (**left**) Scaling of runtime with respect to the number of samples n. (**right**) Scaling of runtime with respect to local error tolerance Tol.

Without adaptive time stepping, numerical error depends on the choice of step size Δt. The runtime is expected to scale as Δt^{-1}. But by setting a local error tolerance we see a better scaling $Tol^{-0.67}$ as seen in Figure A1.

From Figure A2, the runtime of MC SDE simulation scales as $x_0^{0.53}$ with respect to initial position x_0. This is better compared to fixed grid-size FPE solvers where at least an x_0^1 scaling is expected. We also see better than linear runtime scaling with respect to noise levels. Runtime $\sim D^{0.34} \sim \sigma^{0.58}$.

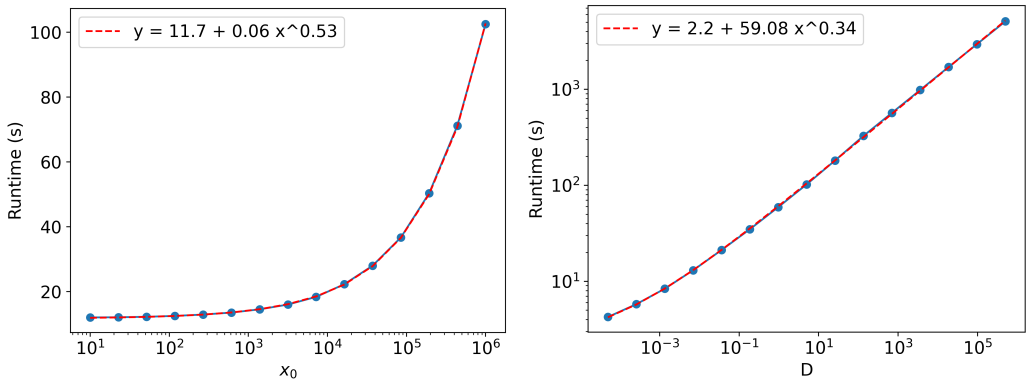

Figure A2. (**left**) Scaling of runtime with respect to different initial position x_0. Note that unlike other plots in this section, the y-axis is not in log scale. (**right**) Scaling of runtime with respect to different noise levels $D := \sigma^2/2$.

Appendix B. Discretization Error

Differentiation can be approximated by a finite difference:

$$f'(t) \approx \frac{f(t+\Delta t) - f(t)}{\Delta t} \tag{A1}$$

Other methods of numerical differentiation exists, but are not suitable for the algorithms in this work. The error in using finite difference to approximate a derivative can be derived by looking at the Taylor expansion.

$$f(t+\Delta t) = f(t) + f'(t)\Delta t + \frac{f''(t)\Delta t^2}{2} + \mathcal{O}(\Delta t^3) \tag{A2}$$

$$\implies f'(t) = \frac{f(t+\Delta t) - f(t)}{\Delta t} - \frac{f''(t)\Delta t}{2} + \mathcal{O}(\Delta t^2) \tag{A3}$$

An integral can be approximate by the Trapezoidal rule:

$$\int_a^b f(t)dt \approx \sum_{k=1}^{N} \frac{f(t_{k-1}) + f(t_k)}{2} \Delta t_k \tag{A4}$$

where the interval (a, b) is divided into N sub-interval and $\Delta t_k := t_k - t_{k-1}$. If the sub-interval have equal width the error in the approximation can be bounded [50].

$$\left| \int_a^b f(t)dt - \sum_{k=1}^{N} \frac{f(t_{k-1}) + f(t_k)}{2} \Delta t_k \right| \leq \max_{a \leq t \leq b} \left| f''(t) \right| \frac{(b-a)^3}{12N^2} \tag{A5}$$

Appendix C. Jensen's Equality

Jensen's inequality states that for a random variable X and convex function φ

$$\varphi(\mathbb{E}[X]) \leq \mathbb{E}[\varphi(X)] \tag{A6}$$

Here we will derive the error term for this inequality. We assume that φ is at least twice differentiable in the domain of interest. Then we can write the Taylor series of φ around $\mu := \mathbb{E}[X]$ with the remainder term as follows:

$$\varphi(X) = \varphi(\mu) + \varphi'(\mu)(X - \mu) + \varphi'(\mu)(X - \mu) + \frac{1}{2!} \int_\mu^X (X - t)\varphi''(t)dt \tag{A7}$$

Taking expectation on both sides we get:

$$\mathbb{E}[\varphi(X)] = \varphi(\mu) + \frac{1}{2!} \mathbb{E}\left[\int_\mu^X (X - t)\varphi''(t)dt \right] \tag{A8}$$

For the case of $\varphi(X) = X^3$ we have:

$$\mathbb{E}\left[X^3\right] = \mu^3 + 3\mathbb{E}\left[\int_\mu^X (X - t)tdt \right] \tag{A9}$$

$$= \mu^3 + \frac{\mathbb{E}\left[X^3\right] - \mu^3}{2} \tag{A10}$$

$$= \mu^3 + \frac{\tilde{\mu}_3 \sigma^3}{2} + \frac{3\mu\sigma^2}{2} \tag{A11}$$

where $\tilde{\mu}_3$ is the skewness of the distribution and σ the standard deviation.

Appendix D. OU Process Exact Solution

For the SDE $dx_t = -\gamma x_t dt + \sqrt{2D}dW_t$, if the initial distribution of states $p(x, 0)$ is given by the following:

$$p(x, 0) = \sqrt{\frac{\beta_0}{\pi}} e^{-\beta_0 (x - x_0)^2} \tag{A12}$$

Then at an arbitrary time time $t > 0$ the probability density if given by [35]:

$$p(x, t) = \sqrt{\frac{\beta(t)}{\pi}} e^{-\beta(t)(x - x_0 e^{-\gamma t})^2} \tag{A13}$$

$$\frac{1}{2\beta(t)} = \frac{e^{-2\gamma t}}{2\beta_0} + \frac{D\left(1 - e^{-2\gamma t}\right)}{\gamma} \tag{A14}$$

We then use *Integrate[]* in Mathematica to evaluate Equation (4), which upon simplification yields the following closed form expression for Γ:

$$\Gamma(t, \gamma, D, x_0, \beta_0) = \sqrt{2}\gamma \sqrt{\frac{\gamma^2 - 4\beta_0 D(\gamma - \beta_0 D) + \gamma\beta_0 x_0^2 (\gamma + 2\beta_0 D(e^{2\gamma t} - 1))}{\left(\gamma + 2\beta_0 D(e^{2\gamma t} - 1)\right)^2}} \tag{A15}$$

This expression can be further integrated to get the Information Length.

$$\mathcal{L}(t,\gamma,D,x_0,\beta_0) = \sqrt{2}\Big[\text{Tanh}^{-1}(\mathcal{G}) - \text{Tanh}^{-1}(\mathcal{W}(t)) + \mathcal{F}\Big(\text{Tanh}^{-1}(\mathcal{G}/\mathcal{F}) - \text{Tanh}^{-1}(\mathcal{W}(t)/\mathcal{F})\Big)\Big] \tag{A16}$$

$$\mathcal{W}(t) = \sqrt{1 + \frac{\gamma\beta_0 x_0^2}{(\gamma - 2\beta_0 D)^2}(\gamma + 2\beta_0 D(e^{2\gamma t} - 1))} \tag{A17}$$

$$\mathcal{G} = \sqrt{1 + \frac{\gamma^2\beta_0 x_0^2}{(\gamma - 2\beta_0 D)^2}} \tag{A18}$$

$$\mathcal{F} = \sqrt{1 + \frac{\gamma\beta_0 x_0^2}{\gamma - 2\beta_0 D}} \tag{A19}$$

A detailed analysis of the problem can be found here [34].

Appendix E. Γ From KL Divergence

KL divergence between two PDFs $p(x)$ and $q(x)$ is defined as:

$$\mathcal{K}[p(x) \mid q(x)] := \int dx\; p(x) \ln\left(\frac{p(x)}{q(x)}\right) \tag{A20}$$

Now consider a time-dependent PDF $p(x,t)$. The equivalence between KL divergence and Γ can be shown as below.

$$\lim_{dt\to 0} \frac{1}{dt^2}\mathcal{K}[p(x,t+dt) \mid p(x,t)] \tag{A21}$$

$$= \lim_{dt\to 0} \frac{1}{dt^2}\int dx\; p(x,t+dt) \ln\left(\frac{p(x,t+dt)}{p(x,t)}\right) \tag{A22}$$

$$= \lim_{dt\to 0} \frac{1}{dt^2}\int dx\; \left[p + (\partial_t p)dt + \frac{1}{2}(\partial_t^2 p)dt^2 + \mathcal{O}\left(dt^3\right)\right] \ln\left[1 + \frac{(\partial_t p)dt}{p} + \frac{1}{2}\frac{(\partial_t^2 p)dt^2}{p} + \mathcal{O}\left(dt^3\right)\right] \tag{A23}$$

$$= \lim_{dt\to 0} \frac{1}{dt^2}\int dx \left[(\partial_t p)dt + \frac{1}{2}\left(\frac{(\partial_t p)^2}{p} + \partial_t^2 p\right)dt^2 + \mathcal{O}\left(dt^3\right)\right] \tag{A24}$$

$$= \lim_{dt\to 0} \frac{1}{2}\int dx\; \frac{(\partial_t p)^2}{p} + \mathcal{O}(dt) = \frac{1}{2}\Gamma^2(t) \tag{A25}$$

Note that from Equation (A23) onwards $p \equiv p(x,t)$ for brevity. To simplify Equation (A24) we use the normalization condition for the PDF $\int dx\; p = 1$, which also leads to the conditions $\int dx\; \partial_t p = 0$ and $\int dx\; \partial_t^2 p = 0$. Using the same arguments, it can be shown that:

$$\lim_{dt\to 0} \frac{1}{dt^2}\mathcal{K}[p(x,t+dt) \mid p(x,t)] = \lim_{dt\to 0} \frac{1}{dt^2}\mathcal{K}[p(x,t) \mid p(x,t+dt)] \tag{A26}$$

References

1. Oksendal, B. *Stochastic Differential Equations: An Introduction with Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
2. Sauer, T. Numerical solution of stochastic differential equations in finance. In *Handbook of Computational Finance*; Springer: Cham, Switzerland, 2012; pp. 529–550.
3. Panik, M.J. *Stochastic Differential Equations: An Introduction with Applications in Population Dynamics Modeling*; John Wiley & Sons: Hoboken, NJ, USA, 2017.
4. Kareem, A.M.; Al-Azzawi, S.N. A stochastic differential equations model for internal COVID-19 dynamics. In Proceedings of the Journal of Physics: Conference Series, Babylon, Iraq, 5–6 December 2020; IOP Publishing: Bristol, UK, 2021; Volume 1818, p. 012121.
5. Mahrouf, M.; Boukhouima, A.; Zine, H.; Lotfi, E.M.; Torres, D.F.; Yousfi, N. Modeling and forecasting of COVID-19 spreading by delayed stochastic differential equations. *Axioms* **2021**, *10*, 18. [CrossRef]

6. El Koufi, A.; El Koufi, N. Stochastic differential equation model of COVID-19: Case study of Pakistan. *Results Phys.* **2022**, *34*, 105218. [CrossRef] [PubMed]
7. Risken, H. Fokker-planck equation. In *The Fokker-Planck Equation*; Springer: Cham, Switzerland, 1996; pp. 63–95.
8. Guel-Cortez, A.J.; Kim, E.j. Information geometric theory in the prediction of abrupt changes in system dynamics. *Entropy* **2021**, *23*, 694. [CrossRef] [PubMed]
9. Amari, S.I.; Nagaoka, H. *Methods of Information Geometry*; American Mathematical Society: Boston, MA, USA, 2000; Volume 191.
10. Gibbs, A.L.; Su, F.E. On choosing and bounding probability metrics. *Int. Stat. Rev.* **2002**, *70*, 419–435. [CrossRef]
11. Majtey, A.; Lamberti, P.W.; Martin, M.T.; Plastino, A. Wootters' distance revisited: A new distinguishability criterium. *Eur. Phys. J. At. Mol. Opt. Plasma Phys.* **2005**, *32*, 413–419. [CrossRef]
12. Diosi, L.; Kulacsy, K.; Lukacs, B.; Racz, A. Thermodynamic length, time, speed, and optimum path to minimize entropy production. *J. Chem. Phys.* **1996**, *105*, 11220–11225. [CrossRef]
13. Ruppeiner, G. Thermodynamics: A Riemannian geometric model. *Phys. Rev. A* **1979**, *20*, 1608. [CrossRef]
14. Gangbo, W.; McCann, R.J. The geometry of optimal transportation. *Acta Math.* **1996**, *177*, 113–161. [CrossRef]
15. Frieden, B.R. *Science from Fisher Information*; Cambridge University Press: Cambridge, UK, 2004; Volume 974.
16. Facchi, P.; Kulkarni, R.; Man'ko, V.; Marmo, G.; Sudarshan, E.; Ventriglia, F. Classical and quantum Fisher information in the geometrical formulation of quantum mechanics. *Phys. Lett. A* **2010**, *374*, 4801–4803. [CrossRef]
17. Itoh, M.; Shishido, Y. Fisher information metric and Poisson kernels. *Differ. Geom. Appl.* **2008**, *26*, 347–356. [CrossRef]
18. Wootters, W.K. Statistical distance and Hilbert space. *Phys. Rev. D* **1981**, *23*, 357. [CrossRef]
19. Braunstein, S.L.; Caves, C.M. Statistical distance and the geometry of quantum states. *Phys. Rev. Lett.* **1994**, *72*, 3439. [CrossRef] [PubMed]
20. Cafaro, C.; Alsing, P.M. Information geometry aspects of minimum entropy production paths from quantum mechanical evolutions. *Phys. Rev. E* **2020**, *101*, 022110. [CrossRef] [PubMed]
21. Hollerbach, R.; Kim, E.j.; Schmitz, L. Time-dependent probability density functions and information diagnostics in forward and backward processes in a stochastic prey–predator model of fusion plasmas. *Phys. Plasmas* **2020**, *27*, 102301. [CrossRef]
22. Kim, E.J.; Hollerbach, R. Time-dependent probability density functions and information geometry of the low-to-high confinement transition in fusion plasma. *Phys. Rev. Res.* **2020**, *2*, 023077. [CrossRef]
23. Kim, E.J. Investigating information geometry in classical and quantum systems through information length. *Entropy* **2018**, *20*, 574. [CrossRef]
24. Kim, E.J.; Hollerbach, R. Geometric structure and information change in phase transitions. *Phys. Rev. E* **2017**, *95*, 062107. [CrossRef]
25. Heseltine, J.; Kim, E.j. Comparing information metrics for a coupled Ornstein–Uhlenbeck process. *Entropy* **2019**, *21*, 775. [CrossRef]
26. Kim, E.J.; Heseltine, J.; Liu, H. Information length as a useful index to understand variability in the global circulation. *Mathematics* **2020**, *8*, 299. [CrossRef]
27. Crooks, G.E. Measuring thermodynamic length. *Phys. Rev. Lett.* **2007**, *99*, 100602. [CrossRef]
28. Feng, E.H.; Crooks, G.E. Far-from-equilibrium measurements of thermodynamic length. *Phys. Rev. E* **2009**, *79*, 012104. [CrossRef] [PubMed]
29. Kim, E.J.; Guel-Cortez, A.J. Causal Information Rate. *Entropy* **2021**, *23*, 1087. [CrossRef] [PubMed]
30. Kim, E.J. Information geometry and non-equilibrium thermodynamic relations in the over-damped stochastic processes. *J. Stat. Mech. Theory Exp.* **2021**, *2021*, 093406. [CrossRef]
31. Kim, E.J. Information Geometry, Fluctuations, Non-Equilibrium Thermodynamics, and Geodesics in Complex Systems. *Entropy* **2021**, *23*, 1393. [CrossRef]
32. Brillouin, L. *Science and Information Theory*; Courier Corporation: Chelmsford, MA, USA, 2013.
33. Kelly, J.L., Jr. A new interpretation of information rate. In *The Kelly Capital Growth Investment Criterion: Theory and Practice*; World Scientific: Singapore, 2011; pp. 25–34.
34. Kim, E.J.; Hollerbach, R. Signature of nonlinear damping in geometric structure of a nonequilibrium process. *Phys. Rev. E* **2017**, *95*, 022137. [CrossRef] [PubMed]
35. Kim, E.j.; Lee, U.; Heseltine, J.; Hollerbach, R. Geometric structure and geodesic in a solvable model of nonequilibrium process. *Phys. Rev. E* **2016**, *93*, 062127. [CrossRef]
36. Scott, D.W.; Sain, S.R. Multidimensional density estimation. *Handb. Stat.* **2005**, *24*, 229–261.
37. Durrett, R. *Probability: Theory and Examples*; Cambridge University Press: Cambridge, UK, 2019; Volume 49.
38. Kloeden, P.E.; Platen, E. Stochastic differential equations. In *Numerical Solution of Stochastic Differential Equations*; Springer: Cham, Switzerland, 1992; pp. 103–160.
39. Mil'shtejn, G. Approximate integration of stochastic differential equations. *Theory Probab. Appl.* **1975**, *19*, 557–562. [CrossRef]
40. Ilie, S.; Jackson, K.R.; Enright, W.H. Adaptive time-stepping for the strong numerical solution of stochastic differential equations. *Numer. Algorithms* **2015**, *68*, 791–812. [CrossRef]
41. Farber, R. *CUDA Application Design and Development*; Elsevier: Amsterdam, The Netherlands, 2011.
42. Thiruthummal, A.A. CUDA Parallel SDE Simulation. 2022. Available online: https://github.com/keygenx/SDE-Sim (accessed on 6 June 2022).

43. Chen, Y.C. A tutorial on kernel density estimation and recent advances. *Biostat. Epidemiol.* **2017**, *1*, 161–187. [CrossRef]
44. Uhlenbeck, G.E.; Ornstein, L.S. On the theory of the Brownian motion. *Phys. Rev.* **1930**, *36*, 823. [CrossRef]
45. Gutiérrez, R.; Gutiérrez-Sánchez, R.; Nafidi, A.; Ramos, E. A diffusion model with cubic drift: Statistical and computational aspects and application to modelling of the global CO_2 emission in Spain. *Environ. Off. Int. Environ. Soc.* **2007**, *18*, 55–69. [CrossRef]
46. Newton, A.P.; Kim, E.J.; Liu, H.L. On the self-organizing process of large scale shear flows. *Phys. Plasmas* **2013**, *20*, 092306. [CrossRef]
47. Kim, E.J.; Hollerbach, R. Time-dependent probability density function in cubic stochastic processes. *Phys. Rev. E* **2016**, *94*, 052118. [CrossRef]
48. Heer, J. Fast & accurate gaussian kernel density estimation. In Proceedings of the 2021 IEEE Visualization Conference (VIS) IEEE, New Orleans, LA, USA, 24–29 October 2021; pp. 11–15.
49. Silverman, B.W. Algorithm AS 176: Kernel density estimation using the fast Fourier transform. *J. R. Stat. Soc. Ser. (Appl. Stat.)* **1982**, *31*, 93–99. [CrossRef]
50. Cruz-Uribe, D.; Neugebauer, C. An elementary proof of error estimates for the trapezoidal rule. *Math. Mag.* **2003**, *76*, 303–306. [CrossRef]