Bayesian Analysis (2025) **TBA**, Number TBA, pp. 1–24

Modelling Pathwise Uncertainty of Stochastic Differential Equations Samplers via Probabilistic Numerics

Yvann Le Fay [*,†], Simo Särkkä[‡], and Adrien Corenflos[§,¶]

Abstract. Probabilistic ordinary differential equation (ODE) solvers have been introduced over the past decade as uncertainty-aware numerical integrators. They typically proceed by assuming a functional prior to the ODE solution, which is then queried on a grid to form a posterior distribution over the ODE solution. As the queries span the integration interval, the approximate posterior solution then converges to the true deterministic one. Gaussian ODE filters, in particular, have enjoyed a lot of attention due to their computational efficiency, the simplicity of their implementation, and their provable fast convergence rates. In this article, we extend the methodology to stochastic differential equations (SDEs) and propose a probabilistic simulator for SDEs. Our approach involves transforming the SDE into a sequence of random ODEs using piecewise differentiable approximations of the Brownian motion. We then apply probabilistic ODE solvers to the individual ODEs, resulting in a pathwise probabilistic solution to the SDE. We establish worst-case strong 1.5 local and 1.0 global convergence orders for a specific instance of our method. We further show how we can marginalise the Brownian approximations, by incorporating its coefficients as part of the prior ODE model, allowing for computing exact transition densities under our model. Finally, we numerically validate the theoretical findings, showcasing reasonable weak convergence properties in the marginalised version.

Keywords: stochastic differential equations, probabilistic numerics, state-space models, Gaussian filters.

MSC2020 subject classifications: 65C30, 60G15.

1 Introduction

Consider the following Itô stochastic differential equation (SDE) with an initial value condition and an additive noise:

$$\begin{aligned} \mathrm{d}X_t &= \mu(X_t, t)\mathrm{d}t + \sigma(t)\mathrm{d}B_t, \qquad t \in [0, T] \\ X_0 &= Z, \end{aligned} \tag{1}$$

where $T > 0$ is a finite deterministic horizon and $\{B_t \ : \ t \in [0, T]\}$ is an m-dimensional Brownian motion defined on some probability space, and Z be a square-integrable ran-

*ENSAE Paris, CREST, Institut Polytechnique de Paris, yvann.lefay@ensae.fr
†Department of Electrical Engineering and Automation, Aalto University
‡Department of Electrical Engineering and Automation, Aalto University, simo.sarkka@aalto.fi
§Department of Electrical Engineering and Automation, Aalto University
¶Department of Statistics, University of Warwick, adrien.corenflos@warwick.ac.uk

© 2025 International Society for Bayesian Analysis https://doi.org/10.1214/25-BA1522

dom variable independent of the canonical filtration generated by B. Additionally, we assume the drift vector function $\mu : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ and the diffusion matrix function $\sigma : [0,T] \to \mathbb{R}^{d\times m}$ are well-behaved functions, ensuring the existence of a unique strong solution X to (1) (see, e.g., Oksendal, 2003, Ch. 5.3).

Given a discretisation $0 = t_0 < t_1 < t_2 < \cdots < t_K = T$, sampling from the finite-dimensional distribution $(X_{t_0}, \dots X_{t_K})$ can then be done by sampling iteratively from the conditional densities $p_{t_k,t_{k+1}}(X_{t_{k+1}}|X_{t_k})$. However, closed-form or otherwise tractable expressions for these densities are rarely known, and we need instead to resort to numerical approximations. The simplest such approach is given by the Euler–Maruyama (EM) scheme (see, e.g., Kloeden, 1992, Ch. 10.2), in which we informally speaking replace infinitesimal differences with finite differences, leveraging the fact that Brownian increments can be simulated exactly. Thus, the solution is approximated by

$$X_{t_{k+1}} \approx X_{t_k} + \mu(X_{t_k}, t_k)(t_{k+1} - t_k) + \sigma(t_k)(B_{t_{k+1}} - B_{t_k}). \tag{2}$$

The EM scheme (2) has both weak and strong convergence rates of order 1, which means that a discrete-time approximation X^δ, where δ is the maximum of the differences $t_{k+1} - t_k$, converges to the true solution X as δ goes to 0. Formally, it converges weakly with order $\beta > 0$, if, for any polynomial function g, $\max_{k\in[0,K]} ||\mathbb{E}(g(X_{t_k})) - \mathbb{E}(g(X^\delta_{t_k}))|| = O(\delta^\beta)$, and it converges strongly with order $\gamma > 0$, if $\max_{k\in[0,K]} \mathbb{E}(||X_{t_k} - X^\delta_{t_k}||) = O(\delta^\gamma)$ (see, e.g., Kloeden, 1992, Ch. 9.6, Ch. 9.7).

Higher-order schemes (for a review, see, e.g., Kloeden, 1992, Ch. 10, Ch. 11) may sometimes be necessary, either for computational reasons (allowing for better precision at a similar cost) or for physical/statistical reasons. For example, when simulating stochastic Hamiltonian dynamics (see, e.g., Burrage and Burrage, 2014), energy preservation of the system is of paramount importance. The need for higher-order schemes is sometimes more subtle, as in Ditlevsen and Samson (2018) who demonstrate that, for instance, the EM scheme fails to capture the covariance of the transition density arising from hypoelliptic diffusions, preventing parameter estimation for such SDEs if a higher-order scheme is not used.

The discretisation errors of the numerical approximation schemes add up a layer of additional numerical uncertainty on top of the desirable structural randomness, that is, the law of the process X one would be interested in recovering. In this article, we aim to provide a model to quantify this numerical uncertainty by extending the probabilistic numerics (PN, Hennig et al., 2022) framework for ODEs to SDEs. At the core, our method relies on modelling these uncertainties in terms of a corrected posterior distribution over the solution. This perspective has recently been introduced for ordinary differential equations (ODEs, see, e.g., Chkrebtii et al., 2016; Tronarp et al., 2019, 2021) as part of the probabilistic numerics framework which we briefly review in Section 2.1. Detailed contributions of the present article are given at the end of this section.

This problem of quantifying uncertainty arising from discretisation in SDEs was raised by Lysy (2016) in a commentary to the article of Chkrebtii et al. (2016), treating ODEs. Lysy (2016) highlights that the non-differentiability of the Brownian path hinders the use of the Bayesian methodology given in Chkrebtii et al. (2016) which relies on

jointly modelling the ODE solution and its first derivative by a Gaussian process. Lysy (2016) suggested two alternative approaches. The first suggestion is to approximate the Brownian motion by its pre-generated increments in an Euler–Maruyama fashion. The obtained algorithm is then an inference algorithm on the increments of the process. However, it leads to asymptotically biased drift and diffusion terms as the discretisation mesh becomes thinner, even for constant drift and diffusion functions. The second alternative is to replace the Brownian motion with a well-chosen differentiable approximating process. Specifically, Lysy (2016) proposes to use the integrated Ornstein-Uhlenbeck process (IOUP) G_t, a stationary Gaussian process with $\mathrm{cov}(\dot{G}_s, \dot{G}_t) = e^{-\theta|t-s|}$, after which (1) becomes

$$\mathrm{d}X_t = \mu(X_t, t)\mathrm{d}t + \sigma(t)\mathrm{d}G_t.$$

This approach resembles that of Wong–Zakai approximations (Wong and Zakai, 1965; Twardowska, 1996), but also presents the same issue: for a choice of $\theta > 0$, the solution will exhibit a non-vanishing bias and, as $\theta \to 0$, the resulting ODE will become stiffer and, hence, harder to solve. Moreover, globally uniform Wong–Zakai approximations are, in general, computationally expensive for low convergence rates (Gyöngy and Stinga, 2013).

As stated in Lysy (2016), a differentiable approximation of the Brownian motion, given that the associated random ODE recovers (1) in a suitable limit, enables us to apply the probabilistic numerics framework to SDEs. In this article, we use a piecewise differentiable approximation of the Brownian motion for this purpose.

Outline and Contributions

1. **Goal:** Our manuscript aims to quantify pathwise uncertainty in SDEs with additive noise. To this end, we generalise the Bayesian probabilistic numerics framework for ODEs to handle SDEs with additive noise.

2. **Method Overview:** Our method relies on (i) assuming a prior distribution over the possible paths of the SDE, like Lysy (2016), and (ii) deriving the posterior distribution thereof associated with a likelihood model corresponding to the path satisfying the SDE.

   (i) **Piecewise Differentiable Brownian Motion Approximation:** Instead of using a global approximation (e.g., the Wong-Zakai approximation), our approach uses piecewise differentiable approximations of the Brownian motion.

   (ii) **Bayesian Filtering and Smoothing:** Our inference process is framed within the Bayesian filtering and smoothing framework (Särkkä and Svensson, 2023), which is commonly used in probabilistic numerics for ODEs (see, e.g., Tronarp et al., 2019).

In summary, our approach builds on Bayesian numerics for ODEs, extending it to SDEs with additive noise, using piecewise approximations of Brownian motion and probabilistic filtering for uncertainty quantification.

The remainder of the article is organised as follows. In Section 2, we review the piecewise approximation approach to SDE solving, in particular the method of Foster et al. (2020), as well as probabilistic ODE solver methods with a focus on the Gaussian filters as given in Tronarp et al. (2019). In Section 3, we show how probabilistic ODE solvers can be applied to random ODE approximations of SDEs. In the specific case of Gaussian filtering applied to the method of Foster et al. (2020), we derive theoretical guarantees in the additive noise setting for the strong convergence of our method. Extensions of our proposed method are discussed in Sections 3.2 and 3.3. Most importantly, we discuss the explicit marginalisation of some auxiliary Gaussian variables introduced as the coefficients of the Brownian approximation. In Section 3.4, we show how posterior uncertainty calibration can be achieved *after the fact* via maximum likelihood estimation for the prior diffusion coefficients Finally, in Section 4, we provide numerical evidence showcasing the statistical performances of our method and its extensions.

2 Background

We now turn to review the two main ingredients of our method, the Bayesian framework to quantify numerical uncertainties for ODEs and the piecewise differentiable approximation of the Brownian motion we use to transform the SDE into a sequence of random ODEs. In Section 3, we will then show how to principally combine Sections 2.2 and 2.3 to perform pathwise uncertainty modelling in SDEs.

2.1 Bayesian Uncertainty Quantification for ODEs

Consider the following initial value problem (IVP),

$$\begin{aligned} y'(t) &= f(y(t), t), \quad \forall t \in [0, T] \\ y(0) &= y_0, \end{aligned} \tag{3}$$

where $f : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d$ is the vector field. Solutions to (3) are rarely expressible in closed form, and one needs to resort to some numerical schemes. Traditional numerical integrators for ODEs such as Euler schemes or Runge-Kutta methods are solely point estimators of the trajectory. They typically come with a numerical error that takes the form $||y^h - y||_\infty = O(h^\alpha)$ where y^h is the approximated solution with step size h. Different approaches to performing numerical analysis of the error exist, and recently proposed methods are given, for example, as randomised integration grids (Lie et al., 2022), black-box error inference via interpolation of pre-existing numerical methods (Teymur et al., 2021), or designing numerical methods that intrinsically carry uncertainty via Bayesian procedures, the origin of which can be traced back to at least Skilling (1992).

A recent outlook (Tronarp et al., 2019) to the Bayesian procedure consists in casting the IVP into a Bayesian inference problem in which y is replaced by a random process Y, and is constrained using the measurement process:

$$\mathcal{Z}(t) = Y^{(1)}(t) - f(Y^{(0)}(t), t). \tag{4}$$

Indeed, in the limiting case when $\mathcal{Z} \equiv 0$, the solution contracts to the true solution y of (3), (see, e.g., Hennig et al., 2022, Ch. 6., Sec. 38.1.1). However, the non-linear dependence of $\mathcal{Z}$ on Y hinders the exact computation of the posterior. Hence, it becomes necessary to resort to some approximation schemes for $\mathcal{Z}$ which are typically sample-based (Chkrebtii et al., 2016; Tronarp et al., 2019) or Gaussian-based (Tronarp et al., 2019, 2021). In Chkrebtii et al. (2016), the prior-to-posterior update is obtained by sampling from the posterior predictive distribution and comparing it to the ODE at hand. This approach leads to expensive routines, typically cubic with respect to the number of time steps, but presents the benefit of performing "exact" inference in the limit of an infinite number of samples. On the other hand, Tronarp et al. (2021) assume a Gauss-Markov process prior for Y, and approximate the posterior distribution through the use of Gaussian (Kalman) filters (see, e.g., Särkkä and Svensson, 2023). In this case, the inference can be performed with a linear complexity in the size of the mesh, but results in biased (Gaussian) posterior representations. The specific merits of the two approaches are thoroughly discussed in the textbook Hennig et al. (2022, Ch. 6., Sec. 38.3.1, 38.4), to which we refer the reader. Throughout this article, we consider the latter method, which we review in the next section, and refer to Hennig et al. (2015); Cockayne et al. (2019), and references within, for a historical review of the field.

2.2 The Gaussian ODE Filters

Tronarp et al. (2019, 2021) propose to use non-linear Gaussian filters to solve the IVP (3). We quickly review their approach as well as existing convergence guarantees. Without loss of generality, we consider the unidimensional case $d = 1$. As mentioned in Section 2.1, conditioning Y on $\mathcal{Z} = 0$ on the whole interval $[0, T]$ contracts the posterior to the true solution of (3) but is intractable in practice. Consequently, we need to restrict ourselves to a finite conditioning grid. Let $0 = t_0 < t_1 < \ldots < t_K = T$ be a regular discrete grid with step size h. The prior on Y is set to be the solution of the following SDE,

$$\begin{aligned} \mathrm{d}Y(t) &= FY(t)\mathrm{d}t + L\mathrm{d}B_t, \quad \forall t \in [0, T] \\ Y(0) &= \mathcal{N}(y_0, P(0)), \end{aligned}$$

where the drift and diffusion matrices are given by

$$F = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & \ldots & \ldots & 0 & 1 \\ c_0 & \ldots & \ldots & \ldots & c_q \end{pmatrix}, \qquad L = \begin{pmatrix} 0 \\ \vdots \\ \eta \end{pmatrix}, \tag{5}$$

for some fixed parameters $(c_0, \ldots, c_q, \eta) \in \mathbb{R}^{q+2}$. The solution of this SDE is a Gauss-Markov process satisfying, for any $t, h \geq 0$, $Y(t+h) \sim \mathcal{N}(A(h)Y(t), Q(h))$, with $A(h) = e^{Fh}$ and $Q(h) = \int_0^h e^{Fs} L L^\top e^{F^\top s} \mathrm{d}s$. We are interested in computing the posterior distributions $Y(t_k) \mid [\mathcal{Z}(s_j) = 0, j = 0, \ldots, k]$, for $k = 0, \ldots, K$. The resulting

state-space model is described by the following transition and likelihood on the measurement process (4),

$$\begin{aligned} Y(t_{k+1}) \mid Y(t_k) &\sim \mathcal{N}(A(h)Y(t_k), Q(h)), \\ \mathcal{Z}(t_k) \mid Y(t_k) &\sim \mathcal{N}(H_1 Y(t_k) - f(H_0 Y(t_k), t_k), R(t_k)), \end{aligned}$$

for $k = 0, \ldots, K$, $H_0 = (1, 0, 0, \ldots, 0)$, $H_1 = (0, 1, 0, \ldots, 0)$, and where a measurement variance R has been added to the measurement process for greater generality. However, conditioning $Y(t_k)$ on $\mathcal{Z}(t_j) = 0$ for $j \leq k$ is in general not feasible because of the non-linearity dependency of the measurement process $\mathcal{Z}$ on Y and we need to resort to approximations. Different such approximation schemes exist, either Gaussian-based or particle filter-based (see, e.g., Chopin and Papaspiliopoulos, 2020; Särkkä and Svensson, 2023). Here, we focus on the extended Kalman filter which provides closed-form Gaussian approximations to the solution.

At time t_k, let $Y(t_k) \mid [\mathcal{Z}(t_j) = 0, j = 0, \ldots, k] \sim \mathcal{N}(m(t_k), P(t_k))$, be given by its approximated filtering posterior. At time t_{k+1}, we have $Y(t_{k+1}) \mid [\mathcal{Z}(t_j) = 0, j = 0, \ldots, k] \sim \mathcal{N}(m^-(t_{k+1}), P^-(t_{k+1}))$ with

$$m^-(t_{k+1}) \coloneqq A(h)m(t_k), \quad P^-(t_{k+1}) \coloneqq A(h)P(t_k)A(h)^\top + Q(h). \tag{6}$$

The updated posterior $Y(t_{k+1}) \mid [\mathcal{Z}(t_j) = 0, j = 0, \ldots, k+1] \sim \mathcal{N}(m(t_{k+1}), P(t_{k+1}))$ can then be approximated by

$$\begin{aligned} S(t_{k+1}) &\coloneqq \tilde{H}(t_{k+1})P^-(t_{k+1})\tilde{H}(t_{k+1})^\top + R(t_{k+1}), \\ K(t_{k+1}) &\coloneqq P^-(t_{k+1})\tilde{H}(t_{k+1})^\top S(t_{k+1})^{-1}, \\ \hat{z}(t_{k+1}) &\coloneqq \tilde{H}m^-(t_{k+1}) - f(H_0 m^-(t_{k+1}), t_{k+1}), \\ m(t_{k+1}) &= m^-(t_{k+1}) - K(t_{k+1})\hat{z}(t_{k+1}), \\ P(t_{k+1}) &= (I_{q+1} - K(t_{k+1})\tilde{H}(t_{k+1}))P^-(t_{k+1}). \end{aligned} \tag{7}$$

Taking $\tilde{H} = H_1$ is the extended Kalman filter of order 0 (EKF0) scheme, while the EKF1 is given by $\tilde{H} : t \mapsto H_1 - J_{f(.,t)}(H_0 m^-(t))H_0$ where $J_{f(.,t)}$ denotes the Jacobian of $f(., t)$ (for a complete review of Gaussian state-space model inference, see, e.g., Särkkä and Svensson, 2023, Ch. 5).

2.3 Approximation of the Brownian Motion for Solving SDEs

As mentioned in Section 1, we are interested in numerically solving the following additive-noise SDE:

$$\mathrm{d}X_t = \mu(X_t, t)\mathrm{d}t + \sigma(t)\mathrm{d}B_t, \qquad t \in [0, T], \tag{8}$$

whilst preserving uncertainty information regarding its path distribution. To apply the existing probabilistic numerics for ODEs to this problem, we will approximate a sample path of (8) by a path that solves a succession of random ODEs. This is done through the use of differentiable approximations of the Brownian motion.

Algorithm 1: A piecewise approximation for simulating a sample path

input : An integration grid $0 = t_0 < t_1 < \cdots < t_K = T$
output: An approximate discrete sample path $\widetilde{X}_{t_0}, \widetilde{X}_{t_1}, \ldots, \widetilde{X}_{t_K}$
Set $\widetilde{X}_{t_0} = Z$
for $k = 0$ **to** $K - 1$ **do**
  Sample an approximation β_k for $t \in [t_k, t_{k+1})$
  Compute $\widetilde{X}_{t_{k+1}}$ using an ODE solver for the $(k+1)$-th IVP (10)
end
return $\widetilde{X}_{t_0}, \ldots, \widetilde{X}_{t_K}$

We adopt the approach of Foster et al. (2020) which uses fixed-degree approximations on an ever-thinner grid and then stitches them together to form a global approximation of the Brownian motion. Such an approximation allows the construction of schemes with a high order of convergence for a low computational budget. Formally, if the random function β_k is a given approximation of the Brownian motion over the interval $[t_k, t_{k+1}]$, $\beta(t) = \sum_k \mathbb{1}_{[t_k, t_{k+1})}(t)\beta_k(t)$ will be a global piecewise approximation of the Brownian motion over the interval $[0, T]$ constructed using those β_k's. Assuming such an approximation for the Brownian motion, we are then interested in solving the following sequence of random ODEs. For $k \in [0, K-1]$,

$$\begin{aligned} \mathrm{d}X_k(t) &= \mu(X_k(t), t)\mathrm{d}t + \sigma(t)\mathrm{d}\beta_k(t), \qquad \text{for all } t \in [t_k, t_{k+1}), \\ X_0(t_0) &= Z, \qquad X_k(t_k) = X_{k-1}(t_k) \text{ if } k \geq 1. \end{aligned} \tag{9}$$

Assuming that approximations $\beta_k(t)$ are differentiable, the above can be written in a more standard form

$$\begin{aligned} \frac{\mathrm{d}X_k(t)}{\mathrm{d}t} &= f_k(X_k(t), t), \qquad \text{for all } t \in [t_k, t_{k+1}), \\ X_0(t_0) &= Z, \qquad X_k(t_k) = X_{k-1}(t_k) \text{ if } k \geq 1, \end{aligned} \tag{10}$$

where we have introduced the $(k+1)$-th random vector field f_k

$$f_k(x, t) = \mu(x, t) + \sigma(t)\frac{\mathrm{d}\beta_k}{\mathrm{d}t}(t), \qquad \forall x \in \mathbb{R}^d, t \in [t_k, t_{k+1}]. \tag{11}$$

Solving the previous sequence of ODEs (9) is in general not feasible in closed-form, and, instead, we need to resort to a numerical ODE solver to compute $\widetilde{X}_{t_1} \approx X_0(t_1), \ldots, \widetilde{X}_{t_K} \approx X_{K-1}(t_K)$. This is summarised in Algorithm 1.

In practice, the choice of the β_k's greatly impacts the numerical efficiency of Algorithm 1, and different choices may lead to different convergence properties or lack thereof. Foster et al. (2020) introduces a family of L^2-optimal polynomial approximation of the Brownian motion that exhibits strong convergence properties: under smoothness assumption on the drift and diffusion functions, the solution of (9) using the second-degree polynomial approximation converges in $\mathcal{L}^2$, globally, both strongly and weakly

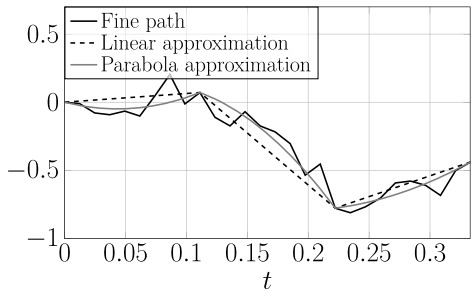


Figure 1: A Brownian path and its linear and parabola approximations. The piecewise derivative of the parabola approximation is discontinuous at the t_k's.

with an order of at least 1.0 and strongly locally with an order of at least 2.0 (Foster et al., 2020, Th. 3.14., Th. 3.17). The parabola approximation is, component-wise, given by

$$\beta_k(t) = B_{t_k} + B_{t_k,t_{k+1}} u + \sqrt{6} u(u-1) I_{t_k,t_{k+1}}, \tag{12}$$

where $u = \frac{t-t_k}{t_{k+1}-t_k}$ and $B_{t_k}, B_{t_k,t_{k+1}}, I_{t_k,t_{k+1}}$ are independent zero-mean Gaussian random variables with variances t_k, $t_{k+1} - t_k$, and $\frac{t_{k+1}-t_k}{2}$, respectively. An illustration of the resulting approximation is given in Figure 1.

3 Modelling Uncertainty in SDEs

In this section, we introduce three novel probabilistic numerics solutions to the additive-noise SDE:

$$\begin{aligned} \mathrm{d}X_t &= \mu(X_t, t)\mathrm{d}t + \sigma(t)\mathrm{d}B_t, \qquad t \in [0, T] \\ X_0 &= Z, \end{aligned} \tag{13}$$

where $T > 0$ is a finite deterministic horizon, μ and σ are vector-valued functions satisfying some latter-specified assumptions, B is a m-dimensional Brownian motion, and Z is a square-integrable random variable independent of the filtration induced by B. The first one, the Gaussian SDE filter, corresponds to applying a single Kalman filter step for each ODE and sampling from the posterior distribution. The second one, the Gaussian Mixture SDE filter, is a slightly modified version of the former where, instead of sampling at each time step, we carry over the mean and the variances of the sample path solution. This then provides the marginal distribution of the solution as a Gaussian with random mean and covariance, justifying the name. The third one, the Marginalised Gaussian SDE filter, incorporates the random coefficients defining the Brownian approximation as part of the latent state modelling the ODE solution. The resulting inference process then forms a joint distribution over the solution and the driving Brownian path, which we can then discard to obtain the marginal ODE

solution. As discussed in Section 2.3, we will consider a, not yet specified, piecewise random ODE given by (9) or equivalently (10).

Importantly, we give strong convergence guarantees for the two first methods when EKF0 (Tronarp et al., 2019) is used on top of the parabola-ODE method (Foster et al., 2020).

3.1 The Gaussian SDE Filters

For the sake of simplicity, we solely focus on the unidimensional problem, $d = 1$, and consider a regular integration grid, $t_{k+1} - t_k = \delta$.

Given a realisation of a Brownian approximation over $[t_k, t_{k+1}]$, our method consists in applying a Gaussian ODE filter to the resulting ODE with vector field a realisation of f_k given by (11). We can then sample from the (Gaussian) approximate solution of the ODE at time t_{k+1} and repeat the procedure between $[t_{k+1}, t_{k+2}]$. Formally, we introduce, for $t \in [t_k, t_{k+1}]$, the variable $Y_k(t) = (Y_k^{(0)}(t), Y_k^{(1)}(t))^\top$ with the initialisation $Y_k(t_k)$ to be sampled from the distribution of $Y_{k-1}^{(0)}(t_k)$:

$$\widetilde{X}_{t_k} \sim \mathcal{L}(Y_{k-1}^{(0)}(t_k)),$$

where $\mathcal{L}(Y_{k-1}^{(0)}(t_k))$ is the (Gaussian) law of $Y_{k-1}^{(0)}(t_k)$. We define the following state-space model,

$$\begin{aligned} Y_k(t_k) &\coloneqq \begin{bmatrix} \widetilde{X}_k \\ f_k(\widetilde{X}_k, t_k) \end{bmatrix}, \\ Y_k(t_{k+1}) \mid Y_k(t_k) &\sim \mathcal{N}(A(\delta) Y_k(t_k), Q(\delta)), \\ \mathcal{Z}_k(t_{k+1}) \mid Y_k(t_{k+1}) &\sim \mathcal{N}(Y_k^{(1)}(t_{k+1}) - f_k(Y_k^{(0)}(t_{k+1}), t_{k+1}), R), \end{aligned}$$

where the observation noise R is assumed to be 0 (see Remarks 1.1, 1.2 and 1.3 in the supplementary material (Le Fay et al., 2025) for a generalisation to $R = O(\delta^2)$). The next value of $\widetilde{X}_{t_k}$ is then sampled from the distribution of $Y_k^{(0)}(t_{k+1}) \mid \{\mathcal{Z}_k(t_{k+1}) = 0\}$ using one of the Gaussian approximations described in Section 2.2, the procedure is then repeated over, resulting in Algorithm 2.

Remark 3.1. *The previous approach is different from solving the piecewise ODE on the whole interval $[0, T]$ using 2.2. Indeed, the discontinuity (see Figure 1) at discretisation points of the piecewise derivative of the parabola approximation $\sum_{k=0}^{K-1} \mathbb{1}_{[t_k, t_{k+1})}(t) \frac{\mathrm{d}\beta_k}{\mathrm{d}s}(t)$ makes the global vector field given by*

$$f(x, t) = \mu(x, t) + \sigma(t) \sum_{k=0}^{K-1} \mathbb{1}_{[t_k, t_{k+1})}(t) \frac{\mathrm{d}\beta_k}{\mathrm{d}s}(t),$$

discontinuous at the t_k's. This sudden change of the vector field at time t_k constrains us to include in the scheme, both the left and right values, $f_{k-1}(\cdot, t_k)$ and $f_k(\cdot, t_k)$, respectively.

Algorithm 2: Gaussian SDE Filter

input : A regular integration grid given by δ
output: An approximate discrete sample path $\widetilde{X}_{t_0}, \widetilde{X}_{t_1}, \ldots, \widetilde{X}_{t_K}$
Set $\widetilde{X}_{t_0} = Z$
for $k = 0$ **to** $K - 1$ **do**
  Sample an approximation β_k for $t \in [t_k, t_{k+1})$
  Set initial distribution $Y_k(t_k) \coloneqq \begin{bmatrix} \widetilde{X}_{t_k} \\ f_k(\widetilde{X}_{t_k}, t_k) \end{bmatrix}$
  Compute the posterior distribution $Y_k(t_{k+1}) \sim \mathcal{N}(m_k(t_{k+1}), P_k(t_{k+1}))$, using one-step EKF0/1
  Sample $\widetilde{X}_{t_{k+1}}$ from the first component of $\mathcal{N}(m_k(t_{k+1}), P_k(t_{k+1}))$
end
return $\widetilde{X}_{t_0}, \ldots, \widetilde{X}_{t_{K-1}}$

Assumption 1 (Regularity of the drift and diffusion functions)**.** *The drift function*

(i) is uniformly bounded with respect to both x and t, i.e., for all $x \in \mathbb{R}$, $t \in [0, T]$, $|\mu(x, t)| \leq L$,

(ii) is continuously differentiable with respect to both x and t,

(iii) admits uniformly bounded partial derivatives with respect to x and t, i.e., for all $x \in \mathbb{R}$ and $t \in [0, T]$, $\max\{|\partial_x \mu(x, t)|, |\partial_t \mu(x, t)|\} \leq L$,

where L is a suitable finite constant. Moreover, the diffusion function is continuously differentiable with respect to time.

Provided the chosen piecewise polynomial approximation is the parabola approximation given by (12), and under the Assumption 1, we can prove that Algorithm 2 converges as $\delta \to 0$.

Theorem 3.1 (Strong convergence of the Algorithm 2 using EKF0 linearisation)**.** *Assume the drift and diffusion functions satisfy Assumption 1, moreover, assume the Brownian approximation β is given by the piecewise parabola (12). Let X denote the solution of the SDE (13). Let $\widetilde{X}$ denote the solution given by Algorithm 2 using EKF0 linearisation with an integrated Ornstein-Uhlenbeck process of parameter $\theta \geq 0$ (see Section 2.2) as a prior. Then for all $n \in [0, K - 1]$:*

$$||X_{t_1} - \widetilde{X}_{t_1}||_{\mathcal{L}^2} = O(\delta^{1.5}), \quad ||X_{t_{n+1}} - \widetilde{X}_{t_{n+1}}||_{\mathcal{L}^2} = O(\delta),$$

where $||\cdot||_{\mathcal{L}^2} = \sqrt{\mathbb{E}[||\cdot||^2]}$ is the $\mathcal{L}^2$-norm and the expectation is taken over the SDE solution's distribution.

The proof is given in the supplementary material **Proof 1** (Le Fay et al., 2025).

Remark 3.2. *While the bounded derivative assumption in Assumption 1 is standard, the bounded-ness of the drift is a strong assumption, which we use to control terms of the form $\mu\partial_x\mu$. We however believe that this could be weakened. Indeed, numerical experiments in 4 are conducted on a model for which it is not verified, but where the scheme still converges empirically.*

3.2 A Gaussian Mixture Solution

A slightly modified version of the previous scheme, perhaps more in spirit with Gaussian ODE filters, consists in keeping track of the posterior distributions $Y_0(t_1) \sim \mathcal{N}(m_0(t_1), P_0(t_1)), \dots, Y_{K-1} \sim \mathcal{N}(m_{K-1}(t_k), P_{K-1}(t_k))$ instead of sampling from it at each iteration. To this end, we enforce the posterior mean and covariance of the solution to the k-th IVP at time t_k, that is $m^{(0)}_{k-1}(t_k)$ and $P_{k-1}(t_k)_{0,0}$, to be the initial mean and covariance of the prior solution to the $(k+1)$-th IVP at time t_k, $m^{(0)}_k(t_k)$ and $P_k(t_k)_{0,0}$, i.e.,

$$\begin{aligned} m^{(0)}_0(0) &= Z, & m^{(0)}_k(t_k) &\coloneqq m^{(0)}_{k-1}(t_k), \\ P_0(t_0)_{0,0} &\coloneqq 0, & P_k(t_k)_{0,0} &\coloneqq P_{k-1}(t_k)_{0,0}. \end{aligned}$$

The vector field components for the mean and covariance are then updated accordingly to the sampled Brownian path on $[t_k, t_{k+1}]$. For EKF1, the approximations are obtained using first-order Taylor series expansion (see, e.g., Särkkä and Svensson, 2023, Ch. 5):

$$\begin{aligned} m^{(1)}_k(t_k) &= \mathbb{E}[f_k(Y^{(0)}_k(t_k), t_k)] \\ &\approx f_k(m^{(0)}_k, t_k), \\ P_k(t_k)_{0,1} &= \mathbb{C}[Y^{(0)}_k(t_k), f_k(Y^{(0)}(t_k), t_k)] \\ &\approx P_k(t_k)_{0,0}\partial_x f_k(m^{(0)}_k(t_k), t_k), \\ P_k(t_k)_{1,1} &= \mathbb{V}[f_k(Y^{(0)}(t_k), t_k)] \\ &\approx P_k(t_k)_{0,0}(\partial_x f_k(m^{(0)}_k(t_k), t_k))^2, \end{aligned} \tag{14}$$

and for EKF0,

$$m^{(1)}_k(t_k) \approx f_k(m^{(0)}_k, t_k), \quad P_k(t_k)_{0,1} \approx 0, \quad P_k(t_k)_{1,1} \approx 0, \tag{15}$$

which essentially does not propagate the uncertainty on the solution to its vector field. This corresponds to Algorithm 3.

Theorem 3.2. *[Strong convergence of Algorithm 3 using EKF0 linearisation] Under the Assumptions of Theorem 3.1, the same convergence orders as 3.1 hold for Algorithm 3 using EKF0 linearisation.*

The proof is given in the supplementary material **Proof 2** (Le Fay et al., 2025).

Algorithm 3: Gaussian Mixture SDE Filter

input : A regular integration grid given by δ
output: A mixture of Gaussian for approximating a discrete sample path $Y_0(t_1), \dots, Y_{K-1}(t_K)$
for $k = 0$ **to** $K-1$ **do**
  Sample an approximation β_k for $t \in [t_k, t_{k+1})$
  Set the initial distribution $Y_k(t_k)$ to $\mathcal{N}(m_k(t_k), P_k(t_k))$ using (14) or (15)
  Compute the posterior distribution $Y_k(t_{k+1}) \sim \mathcal{N}(m_k(t_{k+1}), P_k(t_{k+1}))$, using EKF0/1
end
return $Y_0(t_1), \dots, Y_{K-1}(t_K)$

3.3 Marginalising the Brownian Approximation

In the previous sections, we successively sampled, for $k = 0, \dots, K-1$, an approximation of the Brownian motion, and then, conditionally on the approximation, the corresponding probabilistic solution $X_k(t_{k+1})$ to the random ODE. This two-step approach, although consistent, is wasteful in that it (i) requires sampling several random variables that are then discarded, (ii) does not provide a closed-form solution for the (approximate) transition density. In this section, we take a different approach, which consists in marginalising the state distribution with respect to the "random parabola" explicitly rather than under samples, thus avoiding the sampling procedure of the Brownian approximation required by Algorithms 2 and 3. To do so, we now explicitly assume that the Brownian approximation is given by the parabola approximation of (Foster et al., 2020) which we reviewed in Section 2.3.

Marginalising the Brownian approximation then more simply corresponds to computing the joint posterior distribution over the solution and the Gaussian parabola's coefficients $(B_{k\delta,(k+1)\delta}, I_{k\delta,(k+1)\delta})$ appearing in (12). To this end, let us artificially include the Brownian approximation $(B_{k\delta,(k+1)\delta}, I_{k\delta,(k+1)\delta})$ inside the state variable Y_k,

$$Y_k(t) = \begin{pmatrix} Y_k^{(0)}(t) \\ Y_k^{(1)}(t) \\ B_{k\delta,(k+1)\delta} \\ I_{k\delta,(k+1)\delta} \end{pmatrix}, \quad \forall t \in [t_k, t_{k+1}].$$

Similarly to the first scheme, we initialise the first two components of Y_k using a previously sampled solution $\widetilde{X}_{t_k}$, for $k \geq 1$:

$$Y_k^{(0)}(t_k) \coloneqq \widetilde{X}_{t_k} \sim Y_{k-1}^{(0)}(t_k), \quad Y_k^{(1)}(t_k) \coloneqq f_k(\widetilde{X}_{t_k}, t_k).$$

We construct a specific Kalman filtering step by exploiting the structure of the state-space model with respect to $(B_{k\delta,(k+1)\delta}, I_{k\delta,(k+1)\delta})$ to obtain (see Section 1.4 in the

Algorithm 4: Marginalised Gaussian SDE Filter

input : A regular integration grid given by δ
output: An approximate discrete sample path $\widetilde{X}_{t_0}, \widetilde{X}_{t_1}, \ldots, \widetilde{X}_{t_K}$
Set $\widetilde{X}_{t_0} = Z$
for $k = 0$ **to** $K - 1$ **do**
  Set initial distribution $Y_k(t_k)$ to (16) using $\widetilde{X}_{t_k}$
  Sample $\widetilde{X}_{t_{k+1}}$ from the posterior distribution
  $Y_k(t_{k+1}) \sim \mathcal{N}(m_k(t_{k+1}), P_k(t_{k+1}))$, computed using EKF0/1
end
return $\widetilde{X}_{t_0}, \ldots, \widetilde{X}_{t_K}$

supplementary material (Le Fay et al., 2025))

$$Y_k(t_k) \sim \mathcal{N}\left(\begin{bmatrix} \widetilde{X}_{t_k} \\ \mu(\widetilde{X}_{t_k}, t_k) \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{4}{\delta}\sigma(t_k)^2 & \sigma(t_k) & -\frac{\sqrt{6}}{2}\sigma(t_k) \\ 0 & \sigma(t_k) & \delta & 0 \\ 0 & -\frac{\sqrt{6}}{2}\sigma(t_k) & 0 & \frac{\delta}{2} \end{bmatrix}\right), \tag{16}$$

$$Y_k(t_{k+1}) \mid Y_k(t_k) \sim \mathcal{N}(\bar{A}(\delta)Y_k(t_k), \bar{Q}(\delta)),$$

$$\mathcal{Z}_k(t_{k+1}) \mid Y_k(t_{k+1}) \sim \mathcal{N}(Y_k^{(1)}(t_{k+1}) - \bar{f}_k(Y_k^{(0)}(t_{k+1}), t_{k+1}), R = 0),$$

where $\bar{A}$ and $\bar{Q}$ are given by

$$\bar{A} = \mathrm{Diag}(A, 1, 1),\ \bar{Q} = \mathrm{Diag}(Q, 0, 0),$$

A and Q are defined by (5), and $\bar{f}_k$ is the $(k+1)$-th vector field defined using parabola's coefficients as part of the state, i.e.,

$$\bar{f}_k(Y, t) = \mu(Y^{(0)}, t) + \sigma(t)\left(Y^{(2)} + \frac{\sqrt{6}}{\delta}\left(\frac{2(t - t_k)}{\delta} - 1\right)Y^{(3)}\right),$$

where $Y^{(2)}$ and $Y^{(3)}$ are the Gaussian coefficients for the parabola approximation. The EKF0 scheme is then obtained using (7) with

$$\tilde{H}_k(t) = H_1 - \left(0, 0, \frac{\sigma(t)}{\delta}, \sigma(t)\frac{\sqrt{6}}{\delta}\left(\frac{2(t - t_k)}{\delta} - 1\right)\right),$$

while for the EKF1 scheme, we have

$$\tilde{H}_k(t) = H_1 - \left(\partial_x\mu(H_0 m_k^-(t), t), 0, \frac{\sigma(t)}{\delta}, \sigma(t)\frac{\sqrt{6}}{\delta}\left(\frac{2(t - t_k)}{\delta} - 1\right)\right).$$

The resulting procedure is given by Algorithm 4. Contrary to Algorithms 2 and 3, we did not obtain convergence guarantees for Algorithm 4. This is because incorporating

the Brownian path as part of the solution makes its treatment largely more complicated, with the non-linearity impacting additional parts of the algorithm. However, we empirically observe in Section 4 that it enjoys good weak convergence properties, and, therefore, should be considered as a viable algorithm for uncertainty modelling. By the Rao-Blackwell theorem, if one were able to perform the marginalisation step exactly (which we are not given the non-linearity involved, and so the consequence has to be understood heuristically), such moment estimates exhibit lower variance compared to those obtained via simulation, see, e.g., Särkkä and Svensson (2023)[Ch. 7.5].

3.4 Uncertainty Calibration

The uncertainty calibration can be carried out by choosing the prior, in particular, the drift and diffusion matrices (5). While we restrict ourselves to the set of IOUP processes for simplicity, the procedure we discuss below on the calibration of the diffusion coefficient η can be extended to other priors. The most adopted method for uncertainty calibration in the Bayesian ODE solver literature is the one discussed in Tronarp et al. (2019), in which a post-hoc point estimate on the diffusion coefficient η is obtained under the assumption that the vector field is affine. In the affine SDE setting, this condition is equivalent to assuming the drift function μ is affine with respect to x.

Proposition 3.1 (Quasi maximum-likelihood for the diffusion coefficient)**.** *Assume the drift is an affine function with respect to X, i.e., (8) becomes*

$$\begin{aligned} \mathrm{d}X_t &= (g(t) + f(t)X_t)\mathrm{d}t + \sigma(t)\mathrm{d}B_t, \\ X_0 &= Z. \end{aligned}$$

At time t_k, let $m_k(t_{k+1}), P_k(t_{k+1})$ denote the posterior mean and covariance given by Algorithm 3 with transition parameters (A, Q). Similarly, let $(m'_k(t_{k+1}), P'_k(t_{k+1}))$ be the output of Algorithm 3 but now with η-scaled transition parameters $(A, Q' = \eta^2 Q)$. Then, the posterior means coincide, i.e., $m'_k = m_k$ and the posterior covariances P_k and P'_k are proportional with $P'_k = \eta^2 P_k$. Additionally, denote the predicted mean and covariance of the residual error $Z_k(t_{k+1})$ by $\hat{z}_k(t_{k+1})$ and $S_k(t_{k+1})$. Then, conditionally on the Brownian path and the initialisation, the maximum-likelihood estimate for $\hat{\eta}^2$ given we observe the measurement error $\mathcal{Z}_k(t_{k+1}) = 0$ for $k \in [0, K-1]$ is given by

$$\hat{\eta}^2 = \frac{1}{K} \sum_{k=0}^{K-1} \hat{z}_k(t_{k+1})^\top S_k(t_{k+1})^{-1} \hat{z}_k(t_{k+1}). \tag{17}$$

Other calibration methods including inline calibration allowing time-varying diffusion coefficients are discussed in Bosch et al. (2020).

The calibration uncertainty of a trajectory then goes as follows: (1) use Algorithm 3 to sample from a trajectory, (2) apply (17) to calibrate the posterior uncertainty on the solution.

Remark 3.3. *As outlined in Proposition 3.1 this procedure is an exact MLE for affine drifts only, but is biased otherwise, hence the name quasi maximum-likelihood. Nonetheless, it provides a fast, and principled, way to calibrate uncertainties for the trajectory.*

4 Experiments

Throughout this section, we assess the strong order of convergence for the Algorithms 2 and 3 using EKF0 linearisation (see Theorems 3.1 and 3.2). We empirically measure the weak convergence performances of Algorithms 2, 3 and 4. In addition, we provide an illustration of the calibration method given in Proposition 3.1. The JAX code to reproduce the experiments listed below can be found at the following address: `https://github.com/ylefay/bayesianSDEsolver`.

4.1 Experimental Methodology

The fine solution we will use as a comparison to our approximate solution $\widetilde{X}^\delta$ is obtained using the Euler–Maruyama scheme for a time-step δ^2 and is denoted by X^{em,δ^2}. By simulating parabola approximations (12) with scale δ^2, $(B_{k\delta^2,(k+1)\delta^2}, I_{k\delta^2,(k+1)\delta^2})$, we are able to compute the conditional coefficients $(B_{k\delta}, I_{k\delta}) \mid (B_{k\delta^2,(k+1)\delta^2}, I_{k\delta^2,(k+1)\delta^2})$, which will be used to compute $\widetilde{X}^\delta$ by either Algorithm 2 or Algorithm 3. For the state-space model of Algorithm 4, we compute only weak error estimates: this is because no strong convergence can be computed as the path has been marginalised.

The number of sample paths for each experiment is $N = 10^5$ and the considered step sizes are $\delta \in \{2^{-4}, 2^{-5}, \dots, 2^{-10}\}$. The estimates of the strong local and global errors are given by

$$\hat{\varepsilon}_g = \frac{1}{N}\sum_{k=1}^{N} ||\widetilde{X}^\delta_{T,k} - X^{em,\delta^2}_{T,k}||, \qquad \hat{\varepsilon}_l = \frac{1}{N}\sum_{k=1}^{N} ||\widetilde{X}^\delta_{\delta,k} - X^{em,\delta^2}_{\delta,k}||.$$

The weak error is estimated using the polynomial function $g(X) = XX^\top$,

$$\hat{\varepsilon}_{wg} = \frac{1}{N}\left\| \sum_{k=1}^{N} g(\widetilde{X}^\delta_{T,k}) - g(X^{em,\delta^2}_{T,k}) \right\|.$$

Throughout the experiments, we consider the test model defined by the following SDE

$$\begin{aligned} \mathrm{d}X_0(t) &= 1/\varepsilon(s + X_0(t) - X_0(t)^3 - X_1(t))\mathrm{d}t, \\ \mathrm{d}X_1(t) &= (\alpha + \gamma X_0(t) - X_1(t))\mathrm{d}t + \sigma \mathrm{d}B_t, \\ X(0) &= (0,0)^\top, \end{aligned}$$

where $\varepsilon = 0.1$, $s = 0$, $\gamma = 1.5$, $\alpha = 0.8$, $\sigma = 0.3$, $t \in [0,1]$. This corresponds to the stochastic Fitzhugh–Nagumo (FHN) model, which is a stochastic oscillator used to describe spiking neurons (FitzHugh, 1961; Berglund and Gentz, 2006). Thus, the transition density's covariance showcases a high-order $O(\delta^3)$ term on the first component and high-order schemes are needed to capture the variance structure (i.e., a 1.5 weak-order scheme).

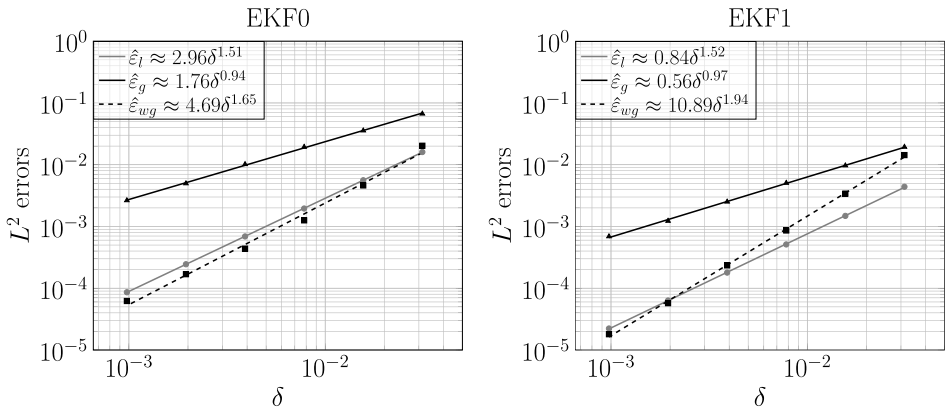


Figure 2: Algorithm 2, Fitzhugh–Nagumo model. Independent integrated Brownian motion (IBM) prior on each coordinates with $\eta = 1$. Strong local and global errors, and weak global errors (grey points, black triangles and black squares, respectively) with log-log regression (grey, black line and dashed black line, respectively).

$\hat{\varepsilon}$	$\hat{a}$	$\hat{b}$	$\hat{c}$	R^2
$\hat{\varepsilon}_{l,\text{EKF0}}$	1.508	1.086	0.007	0.999
$\hat{\varepsilon}_{g,\text{EKF0}}$	0.936	0.565	0.033	0.999
$\hat{\varepsilon}_{wg,\text{EKF0}}$	1.645	1.545	0.216	0.992
$\hat{\varepsilon}_{l,\text{EKF1}}$	1.523	−0.171	0.018	0.999
$\hat{\varepsilon}_{g,\text{EKF1}}$	0.972	−0.586	0.036	0.999
$\hat{\varepsilon}_{wg,\text{EKF1}}$	1.935	2.388	0.081	0.999

Table 1: Linear regression summary of the model (18) for the Algorithm 2 using either EKF0 or EKF1.

4.2 Convergence Rates

We now turn to empirically assessing the convergence orders. Let $\hat{\varepsilon}$ be either the weak error or strong error estimate. In the perfect case where the error estimate is of the form $\hat{\varepsilon} = b\delta^a$, which is the case for a convergence with order α, we have $\ln \hat{\varepsilon} = a \ln \delta + \ln b$. We therefore assume the following linear model with Gaussian residuals:

$$\ln \mathbb{E}[\hat{\varepsilon}] = a \ln \delta + \ln b + \nu, \tag{18}$$

where $\nu \sim \mathcal{N}(0, c^2)$ and perform a least-square regression on our data to estimate a and $\ln b$. The estimators $(\hat{a}, \hat{b})$, as well as the estimated standard deviations $\hat{c}$ and the R^2 coefficients are given in Tables 1, 2 and 3. We draw the following empirical conclusions:

(i) The Algorithms 2 and 3 using EKF0 display 1.5 strong local convergence and 1.0 strong global convergence, confirming Theorems 3.1 and 3.2. This is shown by the solid line regressions in the left plots of Figures 2 and 3.

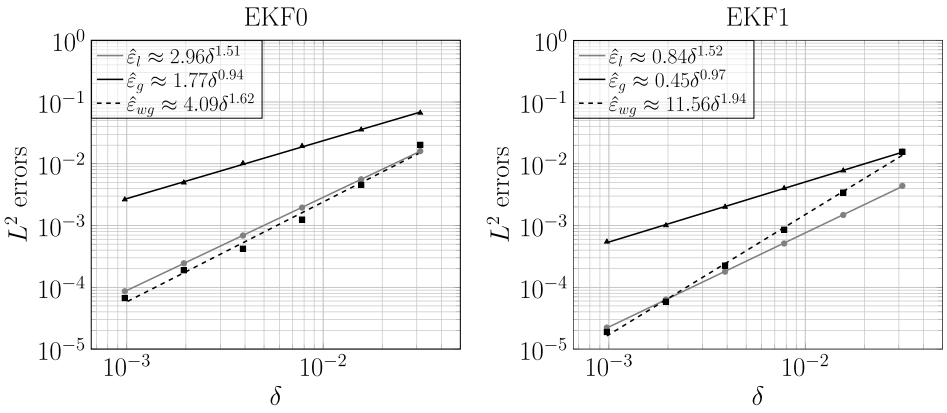


Figure 3: Algorithm 3, Fitzhugh–Nagumo model. Strong, local and global errors, and weak global errors.

$\hat{\varepsilon}$	$\hat{a}$	$\hat{b}$	$\hat{c}$	R^2
$\hat{\varepsilon}_{l,\text{EKF0}}$	1.508	1.086	0.007	0.999
$\hat{\varepsilon}_{g,\text{EKF0}}$	0.937	0.569	0.034	0.999
$\hat{\varepsilon}_{wg,\text{EKF0}}$	1.615	1.409	0.249	0.989
$\hat{\varepsilon}_{l,\text{EKF1}}$	1.523	−0.171	0.0176	0.999
$\hat{\varepsilon}_{g,\text{EKF1}}$	0.973	−0.803	0.026	0.999
$\hat{\varepsilon}_{wg,\text{EKF1}}$	1.943	2.447	0.120	0.998

Table 2: Linear regression summary of the model (18) for Algorithm 3 using either EKF0 or EKF1.

(ii) The above conclusion appears to be valid with EKF1 used instead of EKF0, as shown by the right plots in Figures 2 and 3. The convergence orders are the same but EKF1 gives slightly better performances for the multiplicative factor. See the right plot of Figure 5 for a comparison of the previous schemes as well as the Euler–Maruyama scheme with a step size δ.

(iii) All the above schemes, that is EKF0 and EKF1 versions of Algorithms 2 and 3, exhibit high-order weak convergences, with EKF1 having a higher speed of convergence compared to EKF0. See the dashed line regressions in the right plots of Figures 2, 3, and Figure 5.

(iv) The marginalised version of the parabola-ODE given by Algorithm 4 with EKF0 linearisation exhibits low-order weak convergence, this can be seen with the grey line regression in the left plot of Figure 4. The EKF1 version of Algorithm 4 however does not converge weakly: we observe biased means, see Figure 4.

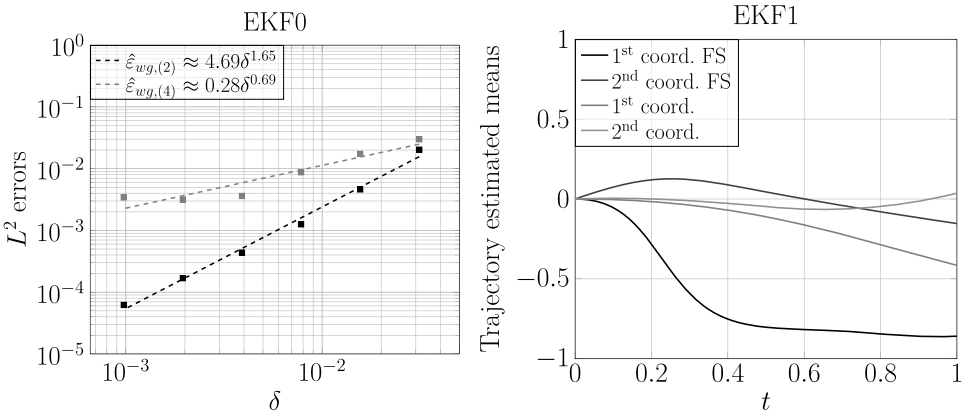


Figure 4: Fitzhugh–Nagumo model. Left: weak global errors for Algorithm 2 and 4 using EKF0, in log-log scale. Right: trajectory means, from the fine solution (FS) and the estimated solution given by Algorithm 4 using EKF1 with $\delta = 2^{-10}$.

$\hat{\varepsilon}$	$\hat{a}$	$\hat{b}$	$\hat{c}$	R^2
$\hat{\varepsilon}_{wg,\text{EKF0}}$	0.694	−1.287	0.350	0.892

Table 3: Linear regression summary of the model (18) for Algorithm 4 using EKF0.

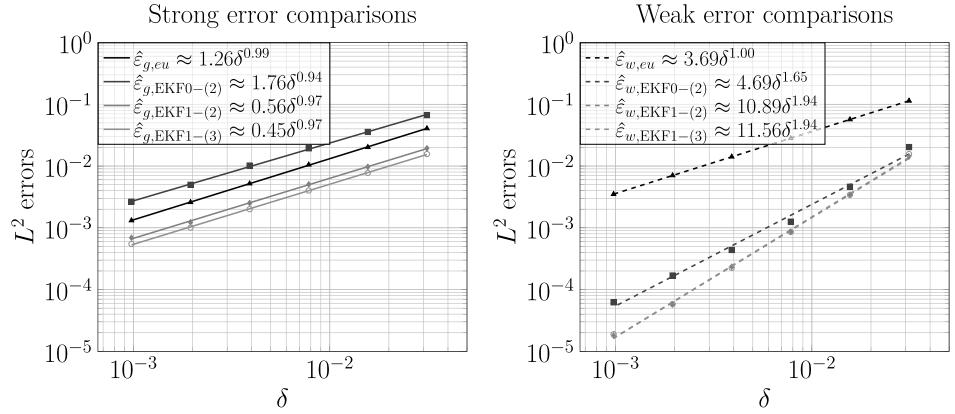


Figure 5: Fitzhugh–Nagumo model. Left: strong global errors for the Euler–Maruyama scheme with parameter δ and Algorithms 2 and 3. Right: similar to the left but with weak global errors.

4.3 Uncertainty Calibration

In this section, we illustrate the calibration uncertainty methodology proposed in Section 3.4. We compute the maximum-likelihood estimate for the diffusion coefficient η^2

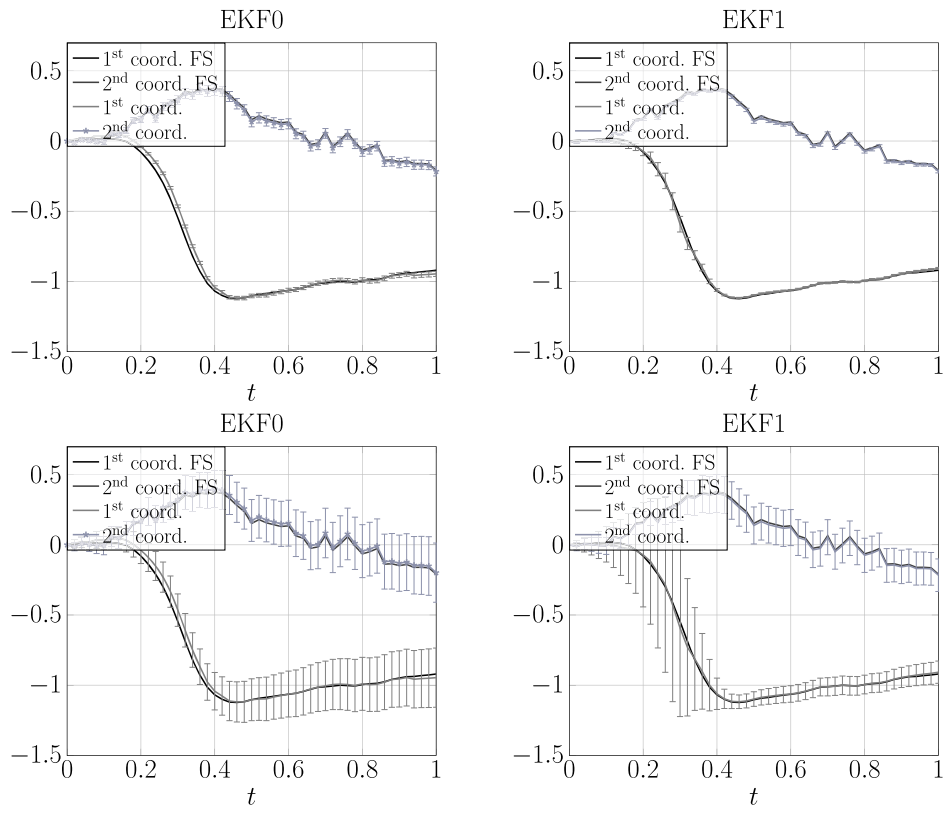


Figure 6: Algorithm 3, Fitzhugh–Nagumo model. Top, prior with $\eta = \sqrt{10}$. Bottom, calibrated prior with $\eta = \hat{\eta}_{\mathrm{MLE}}$ (17). For the two coordinates, the fine solution (FS) sample path obtained via EM and the Gaussian approximation is given by its mean and its standard deviation.

using (17) from Proposition 3.1. We first compute a pathwise solution using Algorithm 3 with $\eta^2 = 1.0$, which is then used to compute the post hoc estimate $\hat{\eta}^2$ (17). The calibrated solution is then obtained by setting $\eta^2 = \hat{\eta}^2$ in Algorithm 3. See Figure 6 for the comparison between the EKF0/1 approximations given by Algorithm 3 with $\eta = \sqrt{10}$ and $\eta = \hat{\eta}_{\mathrm{MLE}}$. The step size is $\delta = 0.02$, the fine solution (FS) denoted by $(X^{\mathrm{em}}_{t_k})_{0 \le k \le K}$ is obtained with the EM scheme with step size δ^2. In order to help with visualisation, in addition, we report the marginal Z-scores associated with both components in Figure 7, computed as follows:

$$\frac{X^{em,(i)}_{t_k} - m^{(i)}_{k-1}(t_k)}{\sqrt{P^{(i,i)}_{k-1}(t_k)}}, \tag{19}$$

for component $i \in \{1, 2\}$ and time step t_k. It can be seen that the non-calibrated posterior solutions (top row) produce errors that oscillate outside their reported uncertainties

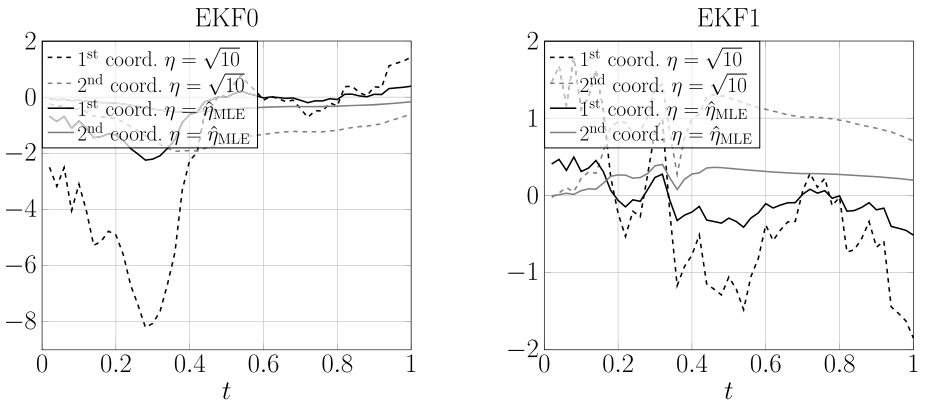


Figure 7: Algorithm 3, Fitzhugh–Nagumo model. Dashed lines, prior with $\eta = \sqrt{10}$. Solid lines, calibrated prior with $\eta = \hat{\eta}_{\text{MLE}}$ (17). For the two coordinates, Z-scores of the fine solution associated with the posterior Gaussian approximation.

while the calibrated ones largely do not, hinting to a better coverage of the posterior uncertainty.

5 Discussion

In this article, we have presented a probabilistic numerics approach to characterising the discretisation error in additive-noise SDEs. We believe that this paves the way for the development of further probabilistic numerics techniques for sampling from SDE solutions. At its core, our methods rely on computing a corrected posterior distribution over the sample path solution of the SDE. We use piecewise differentiable approximations of the Brownian motion on top of which we apply existing Gaussian ODE filtering methods (Tronarp et al., 2019, 2021), allowing for tractable estimation of approximated Gaussian transition densities. To quantify uncertainty, our approach increases computational complexity compared to traditional pointwise SDE simulators, typically costing d^3 due to matrix inversions appearing in the Kalman-filtering routine and potentially (for the EKF1 variant of our method here) requiring gradient evaluations. However, and similarly to Bayesian ODE filters, it can benefit from solutions built either on independence assumptions or Kronecker structures on the model (Krämer et al., 2022), thus making the inference computationally almost as efficient as the EM scheme or other standard SDE samplers.

To summarise, we have introduced three key methods. The first one, the Gaussian SDE filter, consists of drawing a local Brownian approximation, applying a single Kalman filter step for each ODE, and then sampling from the posterior solution. The second one, the Mixture Gaussian SDE filter is a variation of the previous one. Instead of sampling at each time step a solution from the posterior, we propagate the posterior

uncertainties through the Kalman filter, leading to a mixture of Gaussian posteriors given a sample Brownian path. The third one, the Marginalised Gaussian SDE filter, leverages the parabola approximation of the Brownian motion introduced in (Foster et al., 2020) and incorporates the Gaussian coefficients as part of the observed variables, thus marginalising the Brownian path along the SDE solution and resulting in a joint transition density over the solution and the Brownian path.

We have proven empirically and theoretically that our methods retain good pathwise and moment properties. The first two schemes in particular are provably 1.0 strongly convergent in the specific instance where EKF0 is used on top of the parabola approximation and for an IOUP prior. Moreover, numerical experiments highlight a weak convergence order > 1.5 that is yet to be analysed in future work. We anticipate that a similar proof can be adapted for EKF1 (Algorithm 2), which empirically exhibits similar order convergence as EKF0 (respectively strong local order 1.5, strong global order 1, and weak order 2). Moreover, we believe Assumption 1 of bounded drift functions can be weakened: indeed the drift function in the FHN model is not bounded, does not admit bounded derivative, and is not Lipschitz; this did not prevent the method from working in this case. The third, marginalised, scheme appears numerically to be weakly convergent with low-order convergence when EKF0 is used. However, when EKF1 is used to propagate uncertainty on the solution, the resulting trajectory estimate suffers from non-negligible biased moment estimations, likely resulting from an error accumulation that is absent from EKF0 due to its more radical forgetful initialisation.

Perhaps the main drawback of our method lies in its *underconfident* uncertainty estimates, which is a problem inherited from probabilistic numerics for ordinary differential equations, specifically when using EKF1-type linearisations Bosch et al. (2020). The question of better calibrating uncertainty is an active topic of research in the literature on Gaussian-approximated ordinary differential equations (Kersting and Hennig, 2016; Bosch et al., 2020, see, e.g.), and improvements made in this direction are likely to translate immediately to our setting provided the underlying prior/inference methods are compatible with a piecewise approximation approach.

While our current algorithm leverages the specific piecewise Brownian approximation of Foster et al. (2020), it is worth noting that the flexibility of our approach allows for exploring the same framework with different approximations, including higher-order ones, which could lead to performance improvements. Similarly, a large number of extensions of the Gaussian ODE framework have been proposed in the recent literature (see, e.g., Krämer et al., 2022; Bosch et al., 2023), which could be used as a drop-in replacement for EKF0 or EKF1. Furthermore, our Bayesian approach allows for potential improvements of the sampler through the use of adapted solution priors, and uncertainty calibration methods. For example, using periodic priors for periodic SDEs, or Matérn priors for differentiable SDE paths, could enhance the numerical performances of the samplers or, similar to Tronarp et al. (2019), calibrating the prior noise could also lead the SDE sampler to report appropriate uncertainty. Additionally, better Gaussian integrators, such as iterative and implicit methods (Tronarp et al., 2021), are likely directly applicable to our work. We leave the study of their empirical and theoretical properties to future work.

Individual Contributions

The original idea for this article is due to AC. The methodology was primarily developed by both AC and YLF with inputs from SS. The implementation and experimental evaluation are due to YLF with inputs from AC. The theory is due to YLF building on ideas from AC. The original version of this article was written by YLF with feedback and corrections from AC, after which all authors reviewed the manuscript. The project was coordinated by SS.

Acknowledgment

The authors would like to thank two anonymous referees as well as an Associate Editor for their constructive feedback which prompted the discussion on uncertainty calibration and improved the quality of the manuscript overall.

Supplementary Material

Modelling pathwise uncertainty of Stochastic Differential Equations samplers via Probabilistic Numerics (DOI: 10.1214/25-BA1522SUPP; .pdf). The supplementary material contains proofs of Theorems 3.1 and 3.2 in Sections 1.2 and 1.3, respectively, of auxiliary lemmas necessary for these proofs in Section 1.1, and further details regarding the multivariate formulation for the marginalisation procedure of Section 3.3 in Section 1.4.

References

Berglund, N. and Gentz, B. (2006). *Noise-Induced phenomena in slow-fast dynamical systems: A sample-paths approach*, 1 – 16. London: Springer London. MR2197663. 15

Bosch, N., Hennig, P., and Tronarp, F. (2020). "Calibrated Adaptive Probabilistic ODE Solvers." In *International Conference on Artificial Intelligence and Statistics*. 14, 21

Bosch, N., Hennig, P., and Tronarp, F. (2023). "Probabilistic Exponential Integrators." In *Thirty-seventh Conference on Neural Information Processing Systems*. 21

Burrage, P. M. and Burrage, K. (2014). "Structure-preserving Runge-Kutta methods for stochastic Hamiltonian equations with additive noise." *Numerical Algorithms*, 65: 519 – 532. MR3172331. doi: https://doi.org/10.1007/s11075-013-9796-6. 2

Chkrebtii, O. A., Campbell, D. A., Calderhead, B., and Girolami, M. A. (2016). "Bayesian solution uncertainty quantification for differential equations." *Bayesian Analysis*, 11(4): 1239 – 1267. MR3577378. doi: https://doi.org/10.1214/16-BA1017. 2, 5

Chopin, N. and Papaspiliopoulos, O. (2020). *An introduction to Sequential Monte Carlo*. Springer International Publishing. MR4215639. doi: https://doi.org/10.1007/978-3-030-47845-2. 6

Cockayne, J., Oates, C. J., Sullivan, T. J., and Girolami, M. (2019). "Bayesian Probabilistic Numerical Methods." *SIAM Review*, 61(4): 756 – 789. MR4027836. doi: https://doi.org/10.1137/17M1139357. 5

Ditlevsen, S. and Samson, A. (2018). "Hypoelliptic diffusions: Filtering and inference from complete and partial observations." *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 81(2): 361 – 384. MR3928146. 2

FitzHugh, R. (1961). "Impulses and physiological states in theoretical models of nerve membrane." *Biophysical Journal*, 1(6): 445 – 466. 15

Foster, J., Lyons, T., and Oberhauser, H. (2020). "An optimal polynomial approximation of Brownian motion." *SIAM Journal on Numerical Analysis*, 58(3): 1393 – 1421. MR4093364. doi: https://doi.org/10.1137/19M1261912. 4, 7, 8, 9, 12, 21

Gyöngy, I. and Stinga, P. R. (2013). "Rate of convergence of Wong–Zakai approximations for stochastic partial differential equations." In Dalang, R. C., Dozzi, M., and Russo, F. (eds.), *Seminar on Stochastic Analysis, Random Fields and Applications VII*, 95 – 130. Basel: Springer Basel. MR3380097. 3

Hennig, P., Osborne, M. A., and Girolami, M. (2015). "Probabilistic numerics and uncertainty in computations." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179): 20150142. MR3378744. doi: https://doi.org/10.1098/rspa.2015.0142. 5

Hennig, P., Osborne, M. A., and Kersting, H. P. (2022). *Probabilistic Numerics: Computation as machine learning*. Cambridge University Press. MR4420061. doi: https://doi.org/10.1017/9781316681411. 2, 5

Kersting, H. and Hennig, P. (2016). "Active uncertainty calibration in Bayesian ODE solvers." In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'16, 309–318. Arlington, Virginia, USA: AUAI Press. 21

Kloeden, E., Peter E.; Platen (1992). *Numerical solution of stochastic differential equations*. Springer Berlin Heidelberg. MR1214374. doi: https://doi.org/10.1007/978-3-662-12616-5. 2

Krämer, N., Bosch, N., Schmidt, J., and Hennig, P. (2022). "Probabilistic ODE solutions in millions of dimensions." In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 11634 – 11649. 20, 21

Le Fay, Y., Särkkä, S., and Corenflos, A. (2025). "Modelling pathwise uncertainty of Stochastic Differential Equations samplers via Probabilistic Numerics." doi: https://doi.org/10.1214/25-BA1522SUPP. 9, 11, 13

Lie, H. C., Stahn, M., and Sullivan, T. J. (2022). "Randomised one-step time integration methods for deterministic operator differential equations." *Calcolo*, 59(5). MR4386358. doi: https://doi.org/10.1007/s10092-022-00457-6. 4

Lysy, M. (2016). "Comment on article by Chkrebtii, Campbell, Calderhead, and Giro-

lami." *Bayesian Analysis*, 11(4): 1269 – 1273. MR3577379. doi: https://doi.org/10.1214/16-BA1036. 2, 3

Oksendal, B. (2003). *Stochastic differential equations: An introduction with applications*. Berlin, Heidelberg: Springer Berlin Heidelberg. MR2001996. doi: https://doi.org/10.1007/978-3-642-14394-6. 2

Särkkä, S. and Svensson, L. (2023). *Bayesian filtering and smoothing*. Cambridge University Press, second edition. MR4692992. doi: https://doi.org/10.1017/9781108917407. 3, 5, 6, 11, 14

Skilling, J. (1992). *Bayesian Solution of Ordinary Differential Equations*, 23 – 37. Dordrecht: Springer Netherlands. 4

Teymur, O., Foley, C. N., Breen, P. G., Karvonen, T., and Oates, C. J. (2021). "Black box probabilistic numerics." In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*. 4

Tronarp, F., Kersting, H., Särkkä, S., and Hennig, P. (2019). "Probabilistic solutions to ordinary differential equations as nonlinear Bayesian filtering: A new perspective." *Statistics and Computing*. MR4026672. doi: https://doi.org/10.1007/s11222-019-09900-1. 2, 3, 4, 5, 9, 14, 20, 21

Tronarp, F., Särkkä, S., and Hennig, P. (2021). "Bayesian ODE solvers: the maximum a posteriori estimate." *Statistics and Computing*. MR4224948. doi: https://doi.org/10.1007/s11222-021-09993-7. 2, 5, 20, 21

Twardowska, K. (1996). "Wong–Zakai approximations for stochastic differential equations." *Acta Applicandae Mathematica*, 43: 317 – 359. MR1390565. doi: https://doi.org/10.1007/BF00047670. 3

Wong, E. and Zakai, M. (1965). "On the convergence of ordinary integrals to stochastic integrals." *The Annals of Mathematical Statistics*, 36(5): 1560 – 1564. MR0195142. doi: https://doi.org/10.1214/aoms/1177699916. 3