

Probabilistic Solutions To Ordinary Differential Equations As Non-Linear Bayesian Filtering: A New Perspective

Filip Tronarp · Hans Kersting · Simo Särkkä · Philipp Hennig

Received: date / Accepted: date

Abstract We formulate probabilistic numerical approximations to solutions of ordinary differential equations (ODEs) as problems in Gaussian process (GP) regression with non-linear measurement functions. This is achieved by defining the measurement sequence to consist of the observations of the difference between the derivative of the GP and the vector field evaluated at the GP—which are all identically zero at the solution of the ODE. When the GP has a state-space representation, the problem can be reduced to a Bayesian state estimation problem and all widely-used approximations to the Bayesian filtering and smoothing problems become applicable. Furthermore, all previous GP-based ODE solvers, which were formulated in terms of generating synthetic measurements of the vector field, come out as specific approximations. We derive novel solvers, both Gaussian and non-Gaussian, from the Bayesian state estimation problem posed in this paper and com-

pare them with other probabilistic solvers in illustrative experiments.

Keywords Probabilistic Numerics · Initial Value Problems · Non-linear Bayesian Filtering

1 Introduction

We consider an initial value problem (IVP), that is, an ordinary differential equation (ODE)

$$\dot{y}(t) = f(y(t), t), \quad \forall t \in [0, T], \quad y(0) = y_0 \in \mathbb{R}^d, \quad (1)$$

with initial value y_0 and vector field $f: \mathbb{R}^d \times \mathbb{R}_+ \rightarrow \mathbb{R}^d$. Numerical solvers for IVPs (Butcher, 2008) approximate $y: [0, T] \rightarrow \mathbb{R}^d$ and are of paramount importance in almost all areas of science and engineering; but they have—until recently—omitted a probabilistic consideration of the (for all but the most trivial ODEs) inevitable uncertainty from the numerical error over their outputs. Moreover, ODEs are often part of a pipeline surrounded by preceding and subsequent computations which are themselves corrupted by uncertainty from model misspecification, measurement noise, approximate inference or, again, numerical inaccuracy (Kennedy and O’Hagan, 2002). In particular, ODEs often already contain estimated parameters or are integrated afterwards. See Zhang et al (2018) and Chen et al (2018) for recent examples of such computational chains involving ODEs. The field of *probabilistic numerics* (PN) (Hennig et al, 2015) seeks to overcome this ignorance of numerical uncertainty—and the resulting overconfidence—by providing *probabilistic numerical methods*. These solvers quantify numerical errors probabilistically and add them to uncertainty from other sources. Thereby, they can take decisions in a

Filip Tronarp
Aalto University
02150 Espoo, Finland
E-mail: filip.tronarp@aalto.fi

Hans Kersting
University of Tübingen
and Max Planck Institute for Intelligent Systems
72076 Tübingen, Germany
E-mail: hkersting@tuebingen.mpg.de

Simo Särkkä
Aalto University
02150 Espoo, Finland
E-mail: simo.sarkka@aalto.fi

Philipp Hennig
University of Tübingen
and Max Planck Institute for Intelligent Systems
72076 Tübingen, Germany
E-mail: ph@tue.mpg.de

more uncertainty-aware and uncertainty-robust manner (Paul et al, 2018).

In the case of ODEs, one family of probabilistic solvers has first (Skilling, 1991; Hennig and Hauberg, 2014; Schober et al, 2014) treated IVPs as Gaussian process (GP) regression (Rasmussen and Williams, 2006, Chapter 2). Then, Kersting and Hennig (2016) and Schober et al (2018) sped up these methods by regarding them instead as stochastic filtering problems (Øksendal, 2003, Chapter 6). These (completely deterministic) filtering methods converge to the true solution with high polynomial rates (Kersting et al, 2018). The ‘data’ for the Bayesian update is generated via a data generation mechanism (Schober et al, 2018, Section 2.2) by evaluating the vector field f under the current GP estimate of the solution $y(t)$ and linked to the model with a Gaussian likelihood. The inclusion of generated data with another likelihood in a Bayesian framework is discussed by Wang et al (2018, Section 1.2).

The contributions of this paper are the following: Firstly, we circumvent the issue of generating synthetic data, by recasting ODEs as a stochastic filtering problem in a well defined Bayesian state-space model. This enables application of all Bayesian filtering and smoothing techniques to ODEs—as collected, for example, in Särkkä (2013). Secondly, we show how the application of certain inference techniques recovers the previous filtering-based methods. Thirdly, we discuss novel algorithms giving rise to both Gaussian and non-Gaussian solvers. Finally, we present some illustrative experiments demonstrating that these methods are practically useful both for fast inference of the unique solution of an ODE as well as for representing multi-modal distributions of trajectories.

2 Related work: Sampling-Based Solvers

Instead of formulating the solution of Equation (1) as a (Bayesian) GP regression problem, another line of work on probabilistic solvers for ODEs (comprising the methods from Chkrebtii et al 2016; Conrad et al 2017; Teymur et al 2016; Lie et al 2017; Abdulle and Garegnani 2018; Teymur et al 2018) aims to represent the uncertainty arising from the discretization error by a set of samples. While multiplying the computational cost of classical solvers with the amount of samples, these methods can capture arbitrary (non-Gaussian) distributions over the solutions and can reduce overconfidence in inverse problems for ODEs—as demonstrated in Conrad et al (2017, Section 3.2.), Abdulle and Garegnani (2018, Section 7) and Teymur et al (2018). The solvers can be considered as more expensive, but more statistically expressive. This paper contributes a

particle filter as a sampling-based filtering method at the intersection of both lines of work, providing a previously missing link.

3 Bayesian Inference for Initial Value Problems

Formulating an approximation of the solution to Equation (1) at a discrete set of points $\{t_n\}_{n=0}^N$ as a problem of Bayesian inference requires, as always, three things: a prior measure, data, and a likelihood relating the prior to the data.

We start with examining a continuous-time formulation in Section 3.1, where Bayesian conditioning should, in the ideal case, give a Dirac measure at the true solution of Equation (1) as posterior. This has two issues: (1) conditioning on a whole function is not feasible on a computer in finite time and (2) the conditioning operation itself is intractable. Issue (1) is present in classical Bayesian quadrature (Briol et al, 2015) as well. There, the integrand can be evaluated wherever one may wish, but this is infeasible with a bounded computational budget, which is why only a finite number of evaluations of the integrand are used. Issue (2), on the other hand, is unique to Bayesian solutions of ODEs, and turns what is linear Gaussian process regression in Bayesian quadrature into non-linear Gaussian process regression. While this is unfortunate, it appears reasonable that something should be lost as the inference problem is more complex.

With this in mind, a discrete-time non-linear Bayesian filtering problem is posed in Section 3.2, which targets the solution of Equation (1) at a discrete set of points.

3.1 A Continuous-Time Model

Like all previous works mentioned in Section 1, we will consider priors given by a Gaussian process

$$X(t) \sim \text{GP}(\bar{x}, k),$$

where $\bar{x}(t)$ is the mean function and $k(t, t')$ is the covariance function. The vector $X(t)$ is given by

$$X(t) = \left[\left(X^{(1)}(t) \right)^\top, \dots, \left(X^{(q+1)}(t) \right)^\top \right]^\top, \quad (2)$$

where $X^{(1)}(t)$ and $X^{(2)}(t)$ models $y(t)$ and $\dot{y}(t)$, respectively. The remaining $q - 1$ sub-vectors in $X(t)$ can be used to model higher order derivatives of $y(t)$ as in Schober et al (2018); Kersting and Hennig (2016), they can also be used to produce a richer class of prior

measures in general. We define this prior by a stochastic differential equation (Øksendal, 2003), that is

$$X(0) \sim \mathcal{N}(\mu^-(0), \Sigma^-(0)), \quad (3a)$$

$$dX(t) = [FX(t) + u] dt + L dB(t), \quad (3b)$$

where F is the state transition matrix, u is a forcing term, L is a diffusion matrix, and $B(t)$ is a vector of standard Brownian motions.

Note that for $X^{(2)}(t)$ to be the derivative of $X^{(1)}$, F , u and L are such that

$$dX^{(1)}(t) = X^{(2)}(t) dt. \quad (4)$$

The use of an SDE—instead of a generic GP prior—is computationally advantageous because it restricts the priors to Markov processes due to Øksendal (2003, Theorem 7.1.2). This allows for inference with linear time-complexity in N , while the time-complexity is N^3 for Gaussian process priors in general (Hartikainen and Särkkä, 2010).

Inference requires data, and an associated likelihood. Previous authors, such as Schober et al (2018); Chkrebtii et al (2016), put forth the view of the prior measure defining an *inference agent*, which cycles through extrapolating, generating measurements of the vector field, and updating. Here we argue that there is no need for generating measurements, since by re-writing Equation (1) we have the natural requirement

$$\dot{y}(t) - f(y(t), t) = 0. \quad (5)$$

This suggests, that a measurement relating the prior defined by Equation (3) to the solution of Equation (1) ought to be defined as

$$Z(t) = X^{(2)}(t) - f(X^{(1)}(t), t). \quad (6)$$

The measured outcome of $Z(t)$ should of course be $z(t) = 0$. Heuristically, it should hold that

$$E = \left\{ Z(t) = z(t), X^{(1)}(0) = y(0), X^{(2)}(0) = \dot{y}(0) \right\},$$

$$\mathbb{P}[X^{(1)}(t) = y(t) | E] = 1.$$

The conditioning of a random variable on the null set E can be measure-theoretically formalised using the concept of *disintegration* (Cockayne et al, 2017). Though the difference between the development in this paper and that of (Cockayne et al, 2017) is that here the problem is formulated as a standard non-linear Bayesian filtering problem.

3.2 A Discrete-Time Model

In order to make the inference problem tractable, we shall only attempt to condition the process $X(t)$ on $Z(t) = z(t)$ at a set of discrete time-points, $\{t_n\}_{n=0}^N$. Without loss of generality, a uniform grid is considered, $t_{n+1} = t_n + h$. From Equation (3) an equivalent discrete-time system can be obtained (Grewal and Andrews, 2001), hence the inference problem becomes

$$X_0 \sim \mathcal{N}(\mu_0^-, \Sigma_0^-), \quad (8a)$$

$$X_{n+1} | X_n \sim \mathcal{N}(A(h)X_n + \xi(h), Q(h)), \quad (8b)$$

$$Z_n | X_n \sim \mathcal{N}(\dot{C}X_n - f(CX_n, t_n), R(h)), \quad (8c)$$

$$z_n \triangleq 0, \quad n = 1, \dots, N, \quad (8d)$$

where $A(h)$, $\xi(h)$, and $Q(h)$ are the parameters for the discrete time system equivalent to Equation (3), and z_n is the realisation of Z_n . The matrices C and \dot{C} are defined such that $CX_n = X_n^{(1)}$ and $\dot{C}X_n = X_n^{(2)}$, respectively. A measurement variance, $R(h)$, has been added to $Z(t_n)$ for greater generality. This should not affect the posterior contraction towards the true solution if $R(h)$ tends to 0 at a fast enough rate in h .

Note that the inference problem posed in Equation (8) is a standard problem in non-linear Gaussian process regression (Rasmussen and Williams, 2006), also known as filtering and smoothing in stochastic signal processing (Särkkä, 2013). Furthermore, it reduces to Bayesian quadrature when the vector field does not depend on y . This is Proposition 1 below.

Proposition 1 Assume $X_0^{(1)} \sim \delta(x)$, $f(y(t), t) = g(t)$, and $R(h) = 0$. Then the posteriors of $\{X_n^{(1)}\}_{n=1}^N$ are Bayesian quadrature approximations for

$$\int_0^{nh} g(\tau) d\tau, \quad n = 1, \dots, N. \quad (9)$$

Proof See Appendix A.

Remark 1 The Bayesian quadrature method outlined in Proposition 1 conditions on function evaluations outside the domain of integration for $n < N$. This corresponds to the smoothing equations associated with Equation (8). One can show that if the integral on the domain $[0, nh]$ is only conditioned on evaluations of g inside the domain then the filtering estimates associated with Equation (8) are obtained.

3.3 Gaussian Filtering

The inference problem posed in Equation (8) is a standard problem in statistical signal processing and machine learning and the solution is often approximated

by Gaussian filters and smoothers (Särkkä, 2013). In this context the relationships between the filter mean, μ_n^F , and covariance, Σ_n^F with the predictive mean, μ_{n+1}^P , and covariance Σ_{n+1}^P are given by

$$\mu_{n+1}^P = A(h)\mu_n^F + \xi(h), \quad (10a)$$

$$\Sigma_{n+1}^P = A(h)\Sigma_n^F A^\top(h) + Q(h), \quad (10b)$$

which are the prediction equations. The update equations, relating the predictive moments μ_n^P and Σ_n^P with the filter estimate, μ_n^F , and its covariance Σ_n^F , are given by (cf. Särkkä, 2013)

$$S_n = \mathbb{V} \left[\dot{C}X_n - f(CX_n, t_n) \mid z_{1:n-1} \right] + R(h), \quad (11a)$$

$$K_n = \mathbb{C} \left[X_n, \dot{C}X_n - f(CX_n, t_n) \mid z_{1:n-1} \right] S_n^{-1}, \quad (11b)$$

$$\hat{z}_n = \mathbb{E} \left[\dot{C}X_n - f(CX_n, t_n) \mid z_{1:n-1} \right], \quad (11c)$$

$$\mu_n^F = \mu_n^P + K_n(z_n - \hat{z}_n), \quad (11d)$$

$$\Sigma_n^F = \Sigma_n^P - K_n S_n K_n^\top, \quad (11e)$$

where the expectation (\mathbb{E}), covariance (\mathbb{V}) and cross-covariance (\mathbb{C}) operators are with respect to $\mathcal{N}(\mu_n^P, \Sigma_n^P)$. Evaluations of these operators are intractable in general, though various approximation schemes exist in the literature and have been deeply studied. Some standard approximation methods shall be examined below. In particular, we will see that the methods of Schober et al (2018) and Kersting and Hennig (2016) come out as particular approximations to Equation (11).

3.3.1 Taylor-Series Methods

A classical method in the filtering literature to deal with non-linear measurements of the form in Equation (8) is to make a first order Taylor-series expansion, thus turning the problem into a standard update in linear filtering. However, before going through the details of this it is instructive to interpret the method of Schober et al (2018) as an even simpler Taylor-series method. This is Proposition 2 below.

Proposition 2 *Let $R(h) = 0$ and approximate $f(CX_n, t_n)$ by a constant in x_n according to*

$$f(Cx_n, t_n) \approx f(C\mu_n^P, t_n). \quad (12)$$

Then the approximate posterior moments are given by

$$S_n \approx \dot{C}\Sigma_n^P\dot{C}^\top + R(h), \quad (13a)$$

$$K_n \approx \Sigma_n^P\dot{C}^\top S_n^{-1}, \quad (13b)$$

$$\hat{z}_n \approx \dot{C}\mu_n^P - f(C\mu_n^P, t_n), \quad (13c)$$

$$\mu_n^F \approx \mu_n^P + K_n(z_n - \hat{z}_n), \quad (13d)$$

$$\Sigma_n^F \approx \Sigma_n^P - K_n S_n K_n^\top, \quad (13e)$$

which is precisely the solver by Schober et al (2018) since $R(h) = 0$ by assumption.

Proof See Appendix B.

A First Order Approximation. The approximation in Equation (13) can be refined by using a first order approximation, which is known as the extended Kalman filter (EKF) in the signal processing literature (Särkkä, 2013). That is,

$$f(CX_n, t_n) \approx f(C\mu_n^P, t_n) + J_f(C\mu_n^P, t_n)C(X_n - \mu_n^P), \quad (14)$$

where J_f being the Jacobian of f with respect to CX_n . The filter update is then

$$\tilde{C}_n = \dot{C} - J_f(C\mu_n^P, t_n)C, \quad (15a)$$

$$S_n \approx \tilde{C}_n \Sigma_n^P \tilde{C}_n^\top + R(h), \quad (15b)$$

$$K_n \approx \Sigma_n^P \tilde{C}_n^\top S_n^{-1}, \quad (15c)$$

$$\hat{z}_n \approx \dot{C}\mu_n^P - f(C\mu_n^P, t_n), \quad (15d)$$

$$\mu_n^F \approx \mu_n^P + K_n(z_n - \hat{z}_n), \quad (15e)$$

$$\Sigma_n^F \approx \Sigma_n^P - K_n S_n K_n^\top. \quad (15f)$$

Hence the extended Kalman filter computes the residual, $z_n - \hat{z}_n$, in the same manner as Schober et al (2018). However, as the filter gain, K_n , now depends on evaluations of the Jacobian the resulting probabilistic ODE solver is different in general.

While Jacobians of the vector field are seldom exploited in ODE solvers, they play a central role in the exponential Rosenbrock methods (Hochbruck et al, 2009). The Jacobian of the vector field was also recently used by Teymur et al (2018) for developing a probabilistic solver.

3.4 Numerical Quadrature

Another method to approximate the quantities in Equation (11) is by quadrature, which consists of a set of nodes $\{\mathcal{X}_{n,j}\}_{j=1}^J$ with weights $\{w_{n,j}\}_{j=1}^J$ that are associated to the measure $\mathcal{N}(\mu_n^P, \Sigma_n^P)$. These nodes and weights can either be constructed to integrate polynomials up to some order exactly (see e.g. McNamee and Stenger, 1967; Golub and Welsch, 1969), or by Bayesian quadrature (Briol et al, 2015). In either case, the expectation of some function $\psi(X_n)$ is approximated by

$$\mathbb{E}[\psi(X_n)] \approx \sum_{j=1}^J w_{n,j} \psi(\mathcal{X}_{n,j}). \quad (16)$$

Therefore, by appropriate choices of ψ the quantities in Equation (11) can be approximated. For a cross-covariance assumption and a particular choice of quadrature, the method of Kersting and Hennig (2016) is retrieved. This is Proposition 3.

Proposition 3 *Let $\{\mathcal{X}_{n,j}\}_{j=1}^J$ and $\{w_{n,j}\}_{j=1}^J$ be nodes and weights, corresponding to a Bayesian quadrature rule with respect to $\mathcal{N}(\mu_n^P, \Sigma_n^P)$. Furthermore, assume $R(h) = 0$ and that the cross-covariance between $\dot{C}X_n$ and $f(CX_n, t_n)$ vanishes,*

$$\mathbb{C}[\dot{C}X_n, f(CX_n, t_n) \mid z_{1:n-1}] = 0. \quad (17)$$

Then the probabilistic solver proposed in Kersting and Hennig (2016) is a Bayesian quadrature approximation to Equation (11).

Proof See Appendix C.

While a cross-covariance assumption of Proposition 3 reproduces the method of Kersting and Hennig (2016), Bayesian quadrature approximations have previously been used for Gaussian filtering in signal processing applications by Prüher and Šimandl (2015), which in this context gives a new solver.

3.5 Particle Filtering

The Gaussian filtering methods from Section 3.3 may often suffice. However, there are cases where more sophisticated inference methods may be preferable, for instance when the posterior becomes multi-modal due to, for example, bifurcations. While the sampling-based line of probabilistic ODE solvers (introduced in Section 2) could—if sufficient samples are computed—pick up such phenomena, the previous ODE solvers (introduced in Section 1), that (like this paper) recast ODEs as a regression problem with GP prior, fail—due to their restriction to Gaussian distributions—to do so (see Section 1). We overcome this limitation by particle filters, which are based on a sequential formulation of importance sampling (Doucet et al, 2001).

Denote the (intractable) filtering density associated to the inference problem in Equation (8) by $p(x_{0:n} \mid z_{1:n})$. The particle filter operates on a set of particles, $\{X_{n,j}\}_{j=1}^J$, a set of positive weights $\{w_{n,j}\}_{j=1}^J$ associated to the particles that sum to one and an importance density, $g(x_{n+1} \mid x_n, z_n)$. The particle filter then cycles through three steps (1) propagation, (2) re-weighting, and (3) re-sampling (see Särkkä, 2013).

The propagation step then involves sampling particles at $n + 1$ from the importance density:

$$X_{n+1,j} \sim g(x_{n+1} \mid X_{n,j}, z_n). \quad (18)$$

The re-weighting of the particles is done by a likelihood ratio with the product of the measurement density and the transition density of Equation (8), and the importance density. That is, the updated weights are given by

$$w_{n+1,j} \propto \frac{p(z_{n+1} \mid X_{n+1,j}) p(X_{n+1,j} \mid X_{n,j})}{g(X_{n+1,j} \mid X_{n,j}, z_n)}, \quad (19)$$

where there's a proportionality sign to indicate the weights need to be normalised to sum to one after they have been updated according to Equation (19). The weight update is then followed by an optional re-sampling step to keep particle diversity, see Särkkä (2013) for details.

Constructing the Importance Density. In terms of variance, the locally optimal importance density is given by (Doucet et al, 2001)

$$g(x_n \mid x_{n-1}, z_n) \propto p(z_n \mid x_n) p(x_n \mid x_{n-1}). \quad (20)$$

While Equation (20) is almost as intractable as the full filtering distribution, the Gaussian filtering methods from Section 3.3 can be used to make a good approximation. For instance the approximation to the optimal importance density using Equation (13) is given by

$$S_n = \dot{C}Q(h)\dot{C}^\top + R(h), \quad (21a)$$

$$K_n = Q(h)\dot{C}^\top S_n^{-1}, \quad (21b)$$

$$\hat{z}_n = \dot{C}A(h)x_{n-1} - f(CA(h)x_{n-1}, t_n), \quad (21c)$$

$$\mu_n = A(h)x_{n-1} + K_n(z_n - \hat{z}_n), \quad (21d)$$

$$\Sigma_n = Q(h) - K_n S_n K_n^\top, \quad (21e)$$

$$g(x_n \mid x_{n-1}, y_n) = \mathcal{N}(x_n; \mu_n, \Sigma_n). \quad (21f)$$

An importance density can be similarly constructed from Equation (15), resulting in:

$$\tilde{C}_n = \dot{C} - J_f(CA(h)x_{n-1}, t_n) C, \quad (22a)$$

$$S_n \approx \tilde{C}_n Q(h) \tilde{C}_n^\top + R(h), \quad (22b)$$

$$K_n \approx Q(h) \tilde{C}_n^\top S_n^{-1}, \quad (22c)$$

$$\hat{z}_n \approx \dot{C}A(h)x_{n-1} - f(CA(h)x_{n-1}, t_n), \quad (22d)$$

$$\mu_n^F \approx A(h)x_{n-1} + K_n(z_n - \hat{z}_n), \quad (22e)$$

$$\Sigma_n^F \approx Q(h) - K_n S_n K_n^\top, \quad (22f)$$

$$g(x_n \mid x_{n-1}, y_n) = \mathcal{N}(x_n; \mu_n, \Sigma_n). \quad (22g)$$

4 Experimental Results

4.1 The Logistic Equation

In this experiment the logistic equation is considered:

$$\dot{y}(t) = ry(t)(1 - y(t)), \quad y(0) = y_0, \quad t \in [0, 2], \quad (23)$$

which has the solution:

$$y(t) = \frac{\exp(rt)}{1/y_0 - 1 + \exp(rt)}. \quad (24)$$

In the experiments r is set to $r = 3$. We compare the zeroth order solver Equation (13) (Schober et al, 2018) (SCH), the first order solver Equation (15) (EKF), a numerical integration solver based on Proposition 3 (Kersting and Hennig, 2016) (KER), and a numerical integration solver based on approximating Equation (11) (UKF). Both numerical integration approaches use a third degree fully symmetric rule (see McNamee and Stenger, 1967). The filters use a q times integrated Brownian motion prior with $\sigma = 1 \cdot 10^{-1}$ in four different scenarios, $h \in \{3 \cdot 10^{-1}, 3 \cdot 10^{-2}\}$ and $q \in \{1, 2\}$. The prior for all methods is set to match $y(0)$ and $\dot{y}(0)$ exactly, while the higher order derivatives are given a Gaussian zero mean prior with unit covariance.

From Fig. 1 (left column) it can be seen that the solution estimates are fairly similar across experiments. On the other hand, it can be seen that the SCH and KER tend to underestimate the solution in the interval of large growth, most easily seen in Figs. 1f and 1h. While the errors of EKF and UKF tend to be uniformly small over the entire interval.

However, the largest difference appears to be in the uncertainty reported by the different solvers. While the uncertainty of SCH and KER tend to grow over the entire integration domain¹, the uncertainty of EKF and UKF grows in the area where $y(t)$ grows (steep vector field) to then shrink as their estimates reach the equilibrium point at $y(t) = 1$. This can best be seen in Figs. 1b, 1d, 1f, and 1h.

Furthermore, it should be noted that the confidence bands of SCH are not well calibrated for $q = 2$, see Figs. 1h and 1f. The same issue is faced by KER for $q = 2$ and $h = 3 \cdot 10^{-2}$, see Fig 1h. While this can be addressed by selecting a larger σ , the fact that this issue is not present for EKF/UKF suggests they make better use of the available information in this case.

4.2 The FitzHugh-Nagumo Model

The Fitzhugh–Nagumo model is given by:

$$\begin{bmatrix} \dot{y}_1(t) \\ \dot{y}_2(t) \end{bmatrix} = \begin{bmatrix} c \left(y_1(t) - \frac{y_1^3(t)}{3} + y_2(t) \right) \\ -\frac{1}{c} \left(y_1(t) - a + b y_2(t) \right) \end{bmatrix}, \quad (25)$$

¹ Note that the diffusion adaptation proposed by Schober et al (2018, Section 4) is not used here. This, and similar, adaptations can also be used with the proposed Gaussian solvers.

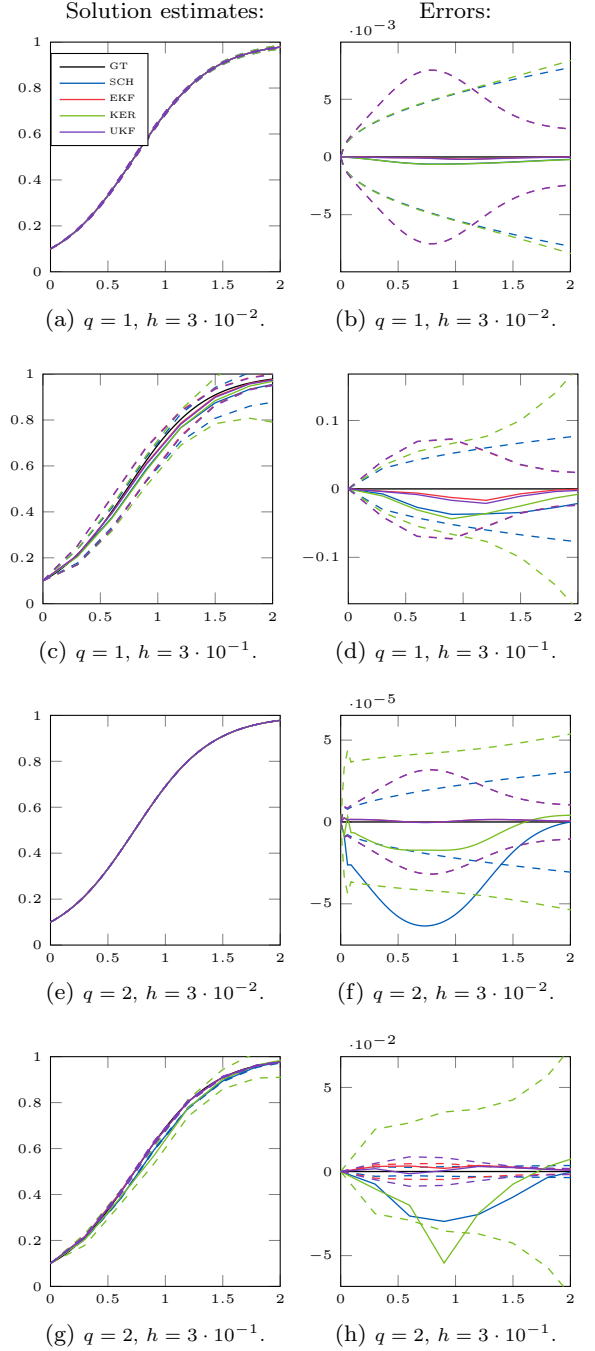


Fig. 1 Solution estimates (left column) and estimation errors (right column) for SCH, KER, EKF, and UKF. 95% confidence bands are given as dashed lines.

where we set $(a, b, c) = (.2, .2, 3)$ and $y(0) = [-1, 1]^T$. We compare SCH, EKF, and UKF using a twice integrated Brownian motion prior ($q = 2$) with $\sigma = 1$ for $h \in \{1 \cdot 10^{-2}, 1 \cdot 10^{-1}\}$. The prior is set to match $y(0)$ and $\dot{y}(0)$ deterministically while $\ddot{y}(0)$ is given a Gaussian prior, $\mathcal{N}(0, 1 \cdot 10^3 \mathbf{I})$. Additionally, a solution

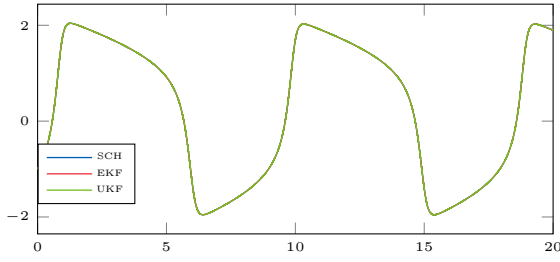
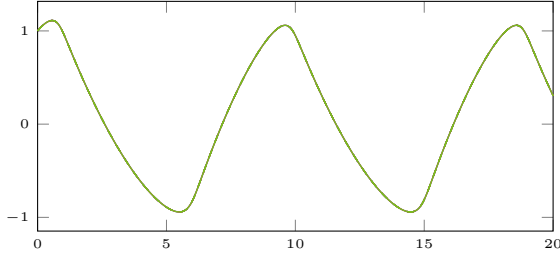

 (a) Estimate of $y_1(t)$.

 (b) Estimate of $y_2(t)$.

Fig. 2 Solution estimates for of the Fitzhugh–Nagumo model for SCH, EKF, and UKF with $q = 2$ and $h = 1 \cdot 10^{-2}$. 95% confidence bands are given as dashed lines.

is computed using a fourth order Runge–Kutta scheme with step-size $h \cdot 10^{-2}$, we take this to be ground truth.

For small step-sizes, $h = 1 \cdot 10^{-2}$, all the methods return similar estimates as can be seen in Fig. 2. On the other hand, the uncertainty reported by the methods is, as in Section 4.1, qualitatively different. While the uncertainty of SCH just expands over the domain of integration, the uncertainty of EKF and UKF exhibit an oscillatory behaviour, expanding in areas where there is rapid change in the solution estimate, to then contract in areas of small change, see Fig. 3. Note that EKF and UKF return very similar estimates and uncertainties why the EKF is difficult to see in the figures.

For the larger step-size, $h = 1 \cdot 10^{-1}$, the EKF and UKF continue to give reasonable estimates while SCH starts behaving erratically at around $t = 2$ and subsequently moves out of phase, see Fig. 4. This invites the hypothesis that EKF and UKF have better stability properties than SCH, in the sense of tolerating larger step-sizes.

4.3 A Bernoulli Equation

In the following experiments we consider a transformation of Equation (23), $\eta(t) = \sqrt{y(t)}$, for $r = 2$. The resulting ODE for $\eta(t)$ now has two stable equilibrium points $\eta(t) = \pm 1$ and an unstable equilibrium point at $\eta(t) = 0$. This makes it a simple test domain for different sampling-based ODE solvers, because different types of posteriors ought to arise. In Section 4.3.1 we

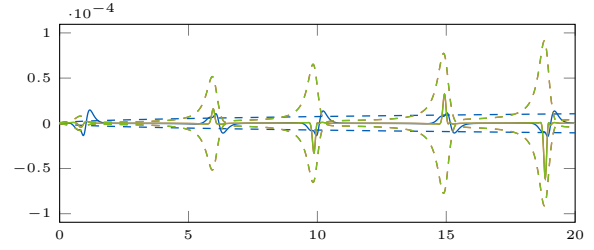
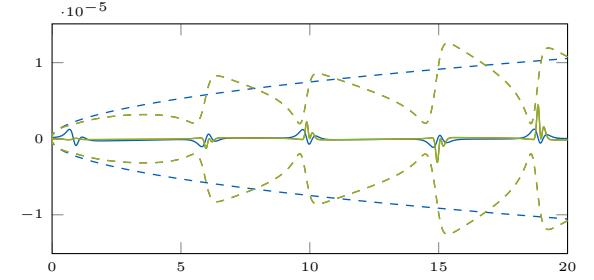

 (a) Error for $y_1(t)$.

 (b) Error for $y_2(t)$.

Fig. 3 Errors for of the Fitzhugh–Nagumo model for SCH, EKF, and UKF with $q = 2$ and $h = 1 \cdot 10^{-2}$. 95% confidence bands are given as dashed lines. The legend is the same as in Fig. 2

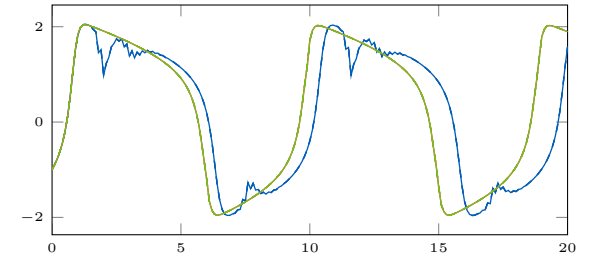
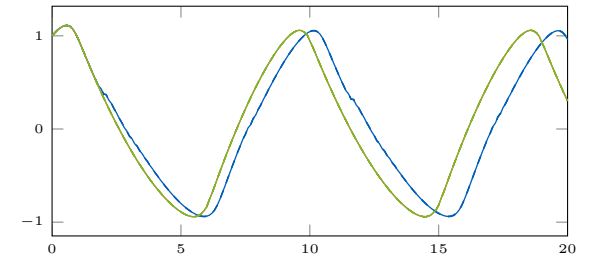

 (a) Estimate of $y_1(t)$.

 (b) Estimate of $y_2(t)$.

Fig. 4 Solution estimates of the Fitzhugh–Nagumo model for SCH, EKF, and UKF with $q = 2$ and $h = 1 \cdot 10^{-1}$. The legend is the same as in Fig. 2

compare Gaussian filters to particle filters for *uncertain* initial conditions on $\eta(t)$. Section 4.3.2 focuses on *certain* initial conditions on $\eta(t)$, and compares particle filtering to the sampling-based solvers of Chkrebtii et al (2016) and Conrad et al (2017) in terms of how uncertainty over trajectories is represented.

4.3.1 Comparing Gaussian Filters to the Particle Filter

We compare the particle filter (PF) without re-sampling, using 5 particles and the proposal density specified by Equation (21) with EKF and UKF. Encoding Gaussian uncertainty on the initial conditions, we set $\eta_0 \sim \mathcal{N}(m_0, 1 \cdot 10^{-3})$, with $m_0 = 1 \cdot 10^{-2}$. The ODE solvers use a $q \in \{1, 2\}$ order integrated Brownian motion with diffusion constant $\sigma = 1$. For $q = 2$, the prior on the initial conditions is set to

$$X_0 \sim \mathcal{N}\left([m_0, f(m_0), J_f(m_0)f(m_0)]^\top, 1 \cdot 10^{-3}\mathbf{I}\right),$$

where $f(\eta)$ is the vector field associated with $\dot{\eta}(t)$ and J_f is its derivative. For $q = 1$ the last dimension is simply removed. Additionally, PF uses $R(h) = h^{2q+1}$ and the EKF/UKF uses $R(h) = 0$. For each q two different step-sizes are compared, $h \in \{3 \cdot 10^{-1}, 3 \cdot 10^{-2}\}$. The ODE is then estimated on the interval $[0, 10]$. We also sample trajectories from the correct measure (GT) by sampling η_0 and plugging it into the exact solution.

The results are shown in Fig. 5. Across all q and h PF appears to sample feasible trajectories as compared to sampling from the correct measure. However, the EKF and UKF have more variation in their behaviour. For $q = 1$ and $h = 3 \cdot 10^{-2}$ The EKF estimate goes toward the stable equilibrium at 1 with a confidence band that sharply widens during the growth of the estimate to then sharply decline as the equilibrium point is reached, see Fig. 5a. On the other hand, the UKF estimate gets stuck close to the unstable equilibrium point while its confidence bands grow to cover almost all feasible trajectories. For $q = 1$ and $h = 3 \cdot 10^{-1}$ the EKF and UKF behave similarly to the UKF for $q = 1$ and $h = 1 \cdot 10^{-2}$, with the notable difference that the confidence band of the EKF expands more than those of the UKF, see Fig. 5b. For $(q, h) = (2, 1 \cdot 10^{-1})$ and $(q, h) = (2, 1 \cdot 10^{-2})$ the EKF and UKF behave similarly to the EKF for $q = 1$ and $h = 1 \cdot 10^{-1}$, but the confidence bands of the UKF neither expand nor contract as rapidly as those of the EKF. Additionally, the UKF estimate is much slower to reach the equilibrium point at 1, see Figs. 5c and 5d.

Furthermore, since $y(t) \rightarrow \pm 1$, $t \rightarrow \infty$, for $y_0 > 0$ and $y_0 < 0$, respectively, the asymptotic (in t) distribution for $y(t)$ is given by

$$y(t) \sim \left\{ \mathbb{P}[y_0 > 0] \delta(y(t) - 1) + \mathbb{P}[y_0 < 0] \delta(y(t) + 1) \right\},$$

which entails the mean and variance of the best Gaussian approximation, in terms of moment matching, are given by $2\mathbb{P}[y_0 > 0] - 1 \approx 0.25$ and $4\mathbb{P}[y_0 > 0]\mathbb{P}[y_0 < 0] \approx 0.94$, respectively. In the context of the present

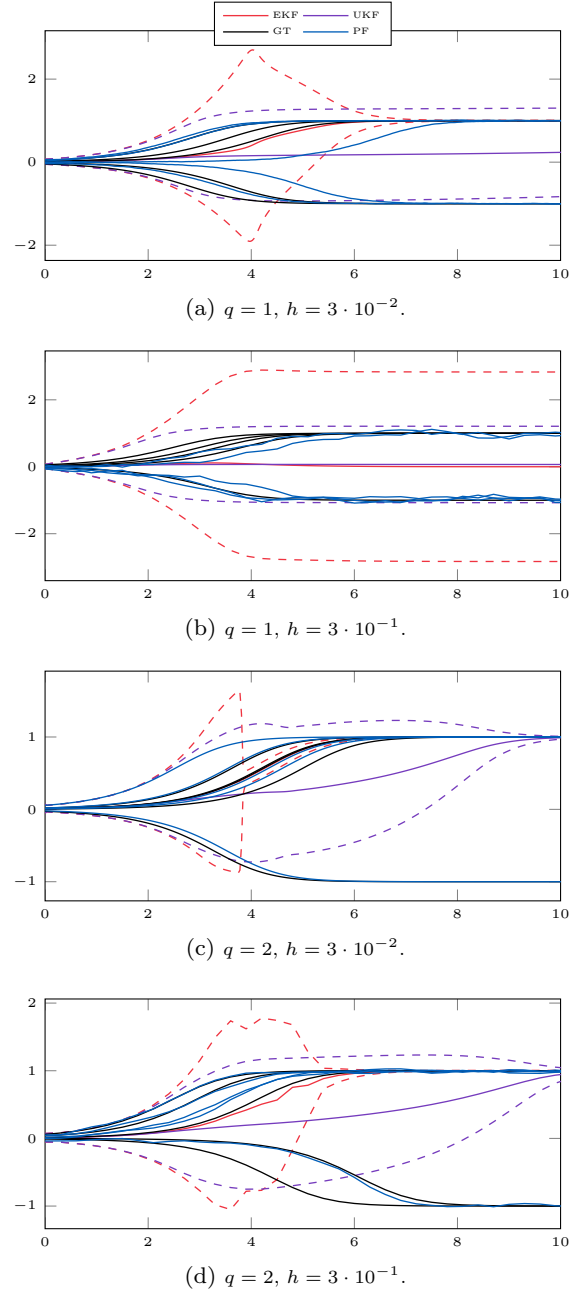


Fig. 5 The estimates of EKF and UKF compared to 5 samples from PF and the true measure. 95% confidence intervals for the EKF and UKF are shown as dashed lines.

experiment that means the UKF performs the best for $q = 1$ and the EKF performs the best for $q = 1$ and $h = 3 \cdot 10^{-1}$. It is possible that σ can be tuned so that EKF/UKF performs well (in the sense of Gaussian approximations to the asymptotic distribution of $y(t)$) for all q and h , however the issue of calibration shall be left for future work.

4.3.2 Sampling-Based Solvers

We compare the proposed particle filter using both the proposal Equation (21) (PF(1)) and EKF proposals (Equation (22)) (PF(2)) with the method by (Chkrebtii et al, 2016) (CHK) and the one by (Conrad et al, 2017) (CON) for estimating $\eta(t)$ on the interval $t \in [0, 5.01]$ with initial condition *deterministically* set to $\eta_0 = 0$. Both PF and CHK use the same prior as in Section 4.3.1, while PF uses a measurement noise of $R(h) = h^{2q+1}$ and CHK uses $R(h) = 0$. CON uses a Runge–Kutta method of order q with perturbation variance h^{2q+1} . The solution is estimated for the same values of q and h as in Section 4.3.1 and the methods are compared in terms of kernel density estimates for the marginal probability distribution of the solution at times $t \in \{0.99, 3, 5.01\}$.

As can be seen in Figs. 6 and 8 for small step-sizes the distributions stay tight around the unstable equilibrium for a fairly long time, particularly for $q = 2$ (better prior). The solver by (Conrad et al, 2017) disperses quicker than the alternatives, followed by the particle filters (see e.g. Figs. 6, 7, and 8). Interestingly the particle filters appear to be the only alternatives producing tri-modal distributions, with two modes going towards both the stable equilibrium points and one staying around the unstable one, see Fig. 6c. Other than that the particle filters behave qualitatively similar to CHK with the particle filters going bi-modal slightly faster.

Additionally, to further contrast the behaviour of the particle filters and CON, five trajectories are simulated from PF2 and CON, see Fig. 10. In addition to escaping the unstable equilibrium point quicker, it can be seen that CON generates rougher paths than PF2.

Lastly, we note that it is not entirely clear how to determine the quality of the solver in this case. If there were uncertainties in ODE due to for example random initial condition then obviously the solver being closest to the correct measure over solutions (in some sense) would be the best option. However, this is not the case here as the ODE is completely deterministic. On the other hand, the solver which contracts to a Dirac at 0 (true solution) as h tends to zero could be a partial measure of quality (in which case CON appears to perform the worst). However, it is to the authors not entirely clear yet how a solver should behave for fixed h in this scenario. We merely note that there are some non-trivial qualitative differences in how different sampling-based solvers represent uncertainty over trajectories.

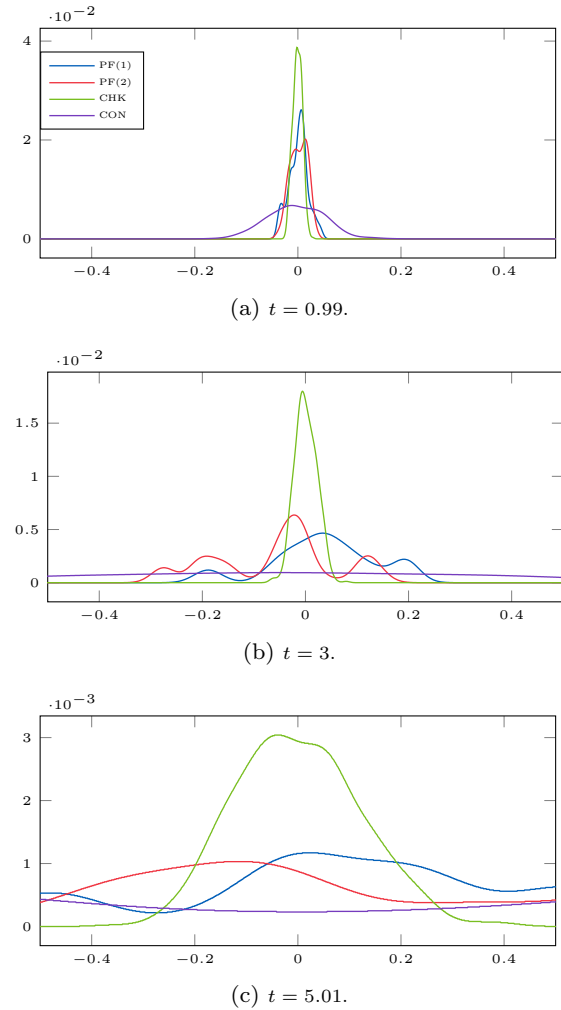


Fig. 6 Kernel density estimates of the distribution of the solution at different time-points for $q = 1$, $h = 3 \cdot 10^{-2}$, and $t \in \{0.99, 3, 5.01\}$.

5 Conclusion and Discussion

In this paper, we have presented a novel formulation of probabilistic numerical solution of ODEs as a standard problem in Gaussian process regression with a non-linear measurement function, and with measurements that are identically zero. Unlike the previous methods, we do not need to rely on synthetic data. Furthermore, as special cases of our method we can recover many of the previously proposed sequential probabilistic ODE solvers. The new formulation enables the derivation of new solvers based on filtering methods from signal processing, where EKF, UKF, and PF have been demonstrated in this paper. More advanced filtering methods can easily be employed in the future, such as the iterative Gaussian filters (Bell and Cathey, 1993; Garcia-Fernandez et al, 2015; Tronarp et al, 2018).

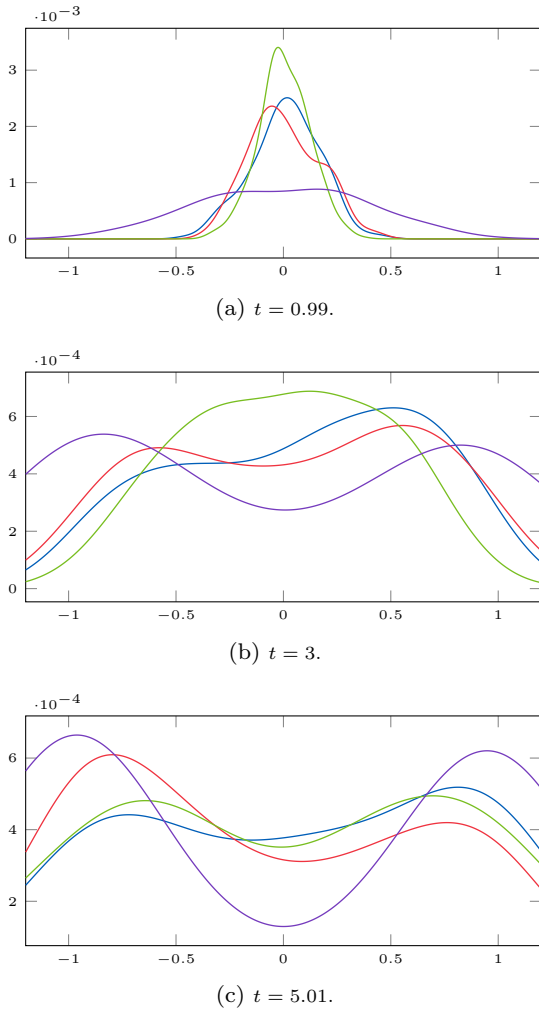


Fig. 7 Kernel density estimates of the distribution of the solution at different time-points for $q = 1$, $h = 3 \cdot 10^{-1}$, and $t \in \{0.99, 3, 5.01\}$.

The theoretical properties of the resulting methods will be analyzed in future work. However, for $h \rightarrow 0$, we expect that the novel Gaussian filters (EKF, UKF) will exhibit polynomial worst-case convergence rates of the mean and its credible intervals, that is, its (Bayesian) uncertainty estimates, as have already been proved in (Kersting et al, 2018) for 0-th order Taylor-series filters with arbitrary constant measurement variance R (see Section 3.3.1).

Our Bayesian recast of ODE solvers might also pave the way toward an average-case analysis of these methods, which has already been executed in (Ritter, 2000) for the special case of Bayesian quadrature. For the PF, we deem a convergence analysis similar to Chkrebtii et al (2016), Conrad et al (2017), Abdulle and Garegnani (2018) and Del Moral (2004). While the understanding of the convergence properties of ODE filters is advancing along these lines, their numerical stability

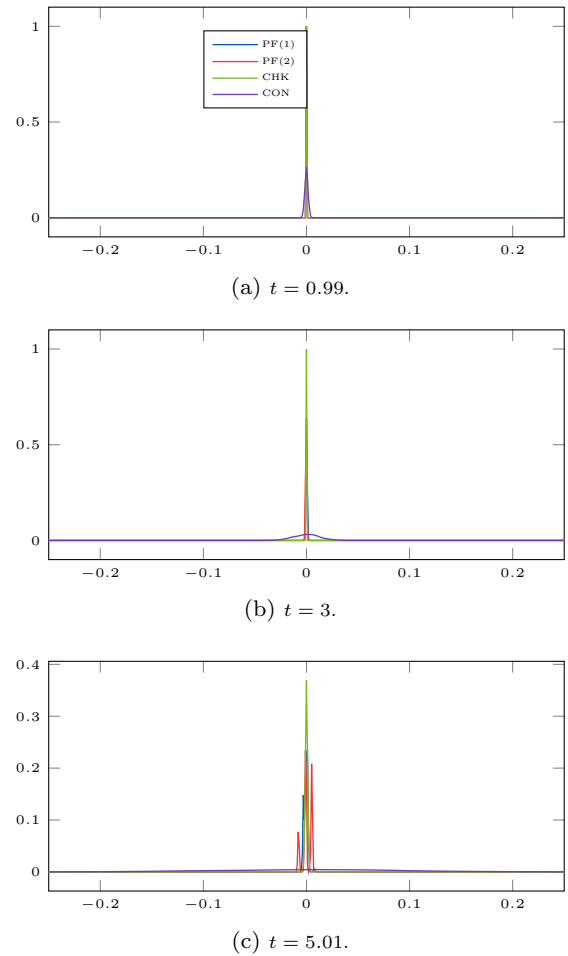


Fig. 8 Kernel density estimates of the distribution of the solution at different time-points for $q = 2$, $h = 3 \cdot 10^{-2}$, $t \in \{0.99, 3, 5.01\}$.

remains unknown. However, the results on spline approximations for ODEs (see, e.g., Loscalzo and Talbot, 1967) might also apply to the present methodology via the correspondence between GP regression and spline function approximations (Kimeldorf and Wahba, 1970).

Acknowledgements This material was developed, in part, at the *Prob Num 2018* workshop hosted by the Lloyd's Register Foundation programme on Data-Centric Engineering at the Alan Turing Institute, UK, and supported by the National Science Foundation, USA, under Grant DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the above-named funding bodies and research institutions.

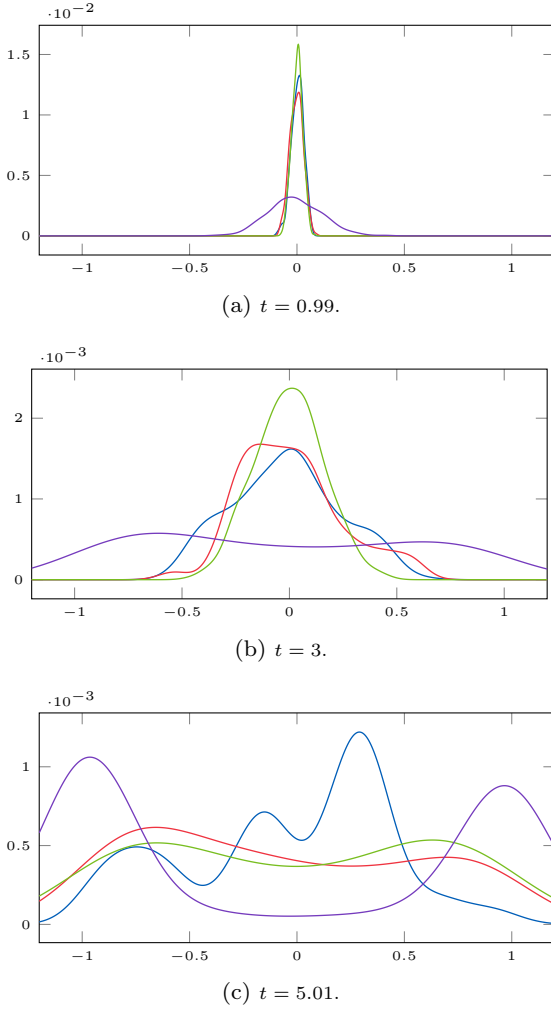


Fig. 9 Kernel density estimates of the distribution of the solution at different time-points for $q = 2$, $h = 3 \cdot 10^{-1}$, $t \in \{0.99, 3, 5.01\}$.

A Proof of Proposition 1

First note that, by Equation (4), we have

$$\frac{d\mathbb{C}[X^{(1)}(t), X^{(2)}(s)]}{dt} = \mathbb{C}[X^{(2)}(t), X^{(2)}(s)]. \quad (26)$$

That is the cross-covariance matrix between $X^{(1)}(t)$ and $X^{(2)}(t)$ is just the integral of the covariance matrix function of $X^{(2)}$. Now define

$$(\mathbf{X}^{(i)})^\top = \left[(X_1^{(i)})^\top \dots (X_N^{(i)})^\top \right], \quad i = 1, \dots, q+1, \quad (27a)$$

$$\mathbf{g}^\top = [g^\top(h) \dots g^\top(Nh)], \quad (27b)$$

$$\mathbf{z}^\top = [z_1^\top \dots z_N^\top]. \quad (27c)$$

Since Equation (3) defines a Gaussian process we have that $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are jointly Gaussian distributed and from Equation (26) the blocks of $\mathbb{C}[\mathbf{X}^{(1)}, \mathbf{X}^{(2)}]$ are given by

$$\mathbb{C}[\mathbf{X}^{(1)}, \mathbf{X}^{(2)}]_{n,m} = \int_0^{nh} \mathbb{C}[X^{(2)}(t), X^{(2)}(mh)] dt$$

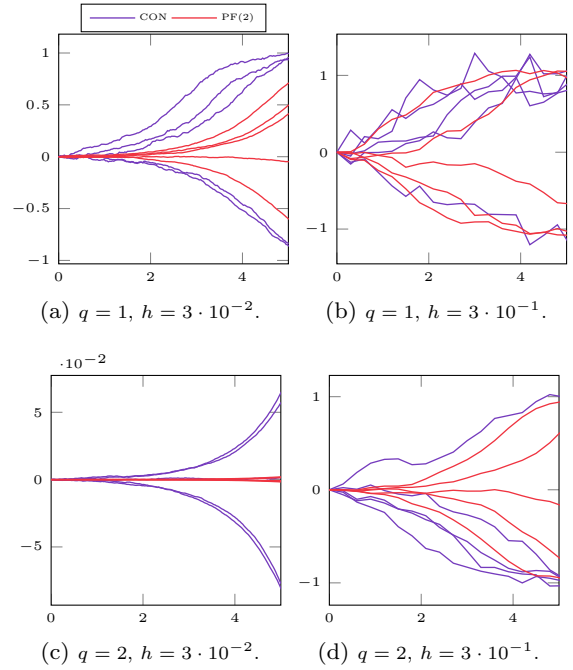


Fig. 10 Five trajectories sampled from CON and PF2.

which is precisely the kernel mean, with respect to the Lebesgue measure on $[0, nh]$, evaluated at mh , see (Briol et al, 2015, Section 2.2). Furthermore,

$$\mathbb{V}[\mathbf{X}^{(2)}]_{n,m} = \mathbb{C}[X^{(2)}(nh), X^{(2)}(mh)],$$

that is, the covariance matrix function (referred to as kernel matrix in Bayesian quadrature literature Briol et al (2015)) evaluated at all pairs in $\{h, \dots, Nh\}$. From Gaussian conditioning rules we have for the posterior means and covariance matrices, denoted by $\mathbb{E}_{\mathcal{D}}[X^{(1)}(nh)]$ and $\mathbb{V}_{\mathcal{D}}[X^{(1)}(nh)]$, respectively, that

$$\begin{aligned} \mathbb{E}_{\mathcal{D}}[X^{(1)}(nh)] &= \mathbb{E}[X^{(1)}(nh)] + \mathbf{w}_n \left(\mathbf{z} + \mathbf{g} - \mathbb{E}[\mathbf{X}^{(2)}] \right) \\ &= \mathbb{E}[X^{(1)}(nh)] + \mathbf{w}_n \left(\mathbf{g} - \mathbb{E}[\mathbf{X}^{(2)}] \right), \end{aligned}$$

$$\mathbb{V}_{\mathcal{D}}[X^{(1)}(nh)] = \mathbb{V}[X^{(1)}(nh)] - \mathbf{w}_n \mathbb{V}[\mathbf{X}^{(2)}] \mathbf{w}_n^\top,$$

where we used the fact that $\mathbf{z} = 0$ by definition and \mathbf{w}_n are the Bayesian quadrature weights associated to the integral of g over the domain $[0, nh]$, given by (see (Briol et al, 2015, Proposition 1))

$$\mathbf{w}_n^\top = \mathbb{V}[\mathbf{X}^{(2)}]^{-1} \begin{bmatrix} \mathbb{C}[X^{(1)}(nh), X^{(2)}(h)]^\top \\ \vdots \\ \mathbb{C}[X^{(1)}(nh), X^{(2)}(Nh)]^\top \end{bmatrix}.$$

B Proof of Proposition 2

As $f(C\mu_n^P, t_n)$ is deterministic, substituting $f(CX_n, t_n)$ for $f(C\mu_n^P, t_n)$ in (11) gives

$$S_n \approx \dot{C}\Sigma_n^P\dot{C}^\top + R(h), \quad (29a)$$

$$K_n \approx \Sigma_n^P\dot{C}^\top S_n^{-1}, \quad (29b)$$

$$\hat{z}_n \approx \dot{C}\mu_n^P - f(C\mu_n^P, t_n), \quad (29c)$$

$$\mu_n^F = \mu_n^P + K_n(z_n - \hat{z}_n), \quad (29d)$$

$$\Sigma_n^F = \Sigma_n^P - K_n S_n K_n^\top, \quad (29e)$$

setting $R(h) = 0$ and recalling that $z_n \triangleq 0$ concludes the proof.

C Proof of Proposition 3

Expand the expressions for S_n and K_n as given by Equation (11) in the paper:

$$\begin{aligned} S_n &= \dot{C}\Sigma_n^P\dot{C}^\top + \mathbb{V}[f(CX_n, t_n) \mid z_{1:n-1}] \\ &\quad - \dot{C}\mathbb{C}[X_n, f(CX_n, t_n) \mid z_{1:n-1}] \\ &\quad - \mathbb{C}[X_n, f(CX_n, t_n) \mid z_{1:n-1}]^\top \dot{C}^\top \\ &= \dot{C}\Sigma_n^P\dot{C}^\top + \mathbb{V}[f(CX_n, t_n) \mid z_{1:n-1}] \\ K_n &= \left(\Sigma_n^P\dot{C}^\top - \mathbb{C}[X_n, f(CX_n, t_n) \mid z_{1:n-1}] \right) S_n^{-1} \\ &= \Sigma_n^P\dot{C}^\top S_n^{-1}. \end{aligned}$$

Where in the second steps the assumption $\mathbb{C}[X_n, f(CX_n, t_n) \mid z_{1:n-1}] = 0$ was used. Lastly, recall that $z_n \triangleq 0$, hence the update equations become

$$S_n = \dot{C}\Sigma_n^P\dot{C}^\top + \mathbb{V}[f(CX_n, t_n) \mid z_{1:n-1}], \quad (31a)$$

$$K_n = \Sigma_n^P\dot{C}^\top S_n^{-1}, \quad (31b)$$

$$\mu_n^F = \mu_n^P + K_n \left(\mathbb{E}[f(CX_n, t_n) \mid z_{1:n-1}] - \dot{C}\mu_n^P \right), \quad (31c)$$

$$\Sigma_n^F = \Sigma_n^P - K_n S_n K_n^\top. \quad (31d)$$

Hence, when $\mathbb{E}[f(CX_n, t_n) \mid z_{1:n-1}]$ and $\mathbb{V}[f(CX_n, t_n) \mid z_{1:n-1}]$ are approximated by Bayesian quadrature, the method of Kersting and Hennig (2016) is obtained.

References

- Abdulle A, Garegnani G (2018) Random time step probabilistic methods for uncertainty quantification in chaotic and geometric numerical integration. arXiv:170303680 [mathNA]
- Bell BM, Cathey FW (1993) The iterated Kalman filter update as a Gauss–Newton method. IEEE Transactions on Automatic Control 38(2):294–297
- Briol FX, Oates CJ, Girolami M, Osborne MA, Sejdinovic D (2015) Probabilistic integration: A role for statisticians in numerical analysis? arXiv:151200933 [statML]
- Butcher JC (2008) Numerical Methods for Ordinary Differential Equations, 2nd edn. John Wiley & Sons, Inc.
- Chen R, Rubanova Y, Bettencourt J, Duvenaud D (2018) Neural ordinary differential equations. In: Advances in Neural Information Processing Systems (NIPS)
- Chkrebtii OA, Campbell DA, Calderhead B, Girolami MA (2016) Bayesian solution uncertainty quantification for differential equations. Bayesian Analysis 11(4):1239–1267
- Cockayne J, Oates C, Sullivan T, Girolami M (2017) Bayesian probabilistic numerical methods. arXiv:170203673 [statME]
- Conrad PR, Girolami M, Särkkä S, Stuart A, Zygalakis K (2017) Statistical analysis of differential equations: introducing probability measures on numerical solutions. Statistics and Computing 27(4):1065–1082
- Del Moral P (2004) Feynman–Kac Formulae: Genealogical and Interacting Particle Systems with Applications. Springer
- Doucet A, De Freitas N, Gordon N (2001) An introduction to sequential Monte Carlo methods. In: Sequential Monte Carlo methods in practice, Springer, pp 3–14
- Garcia-Fernandez AF, Svensson L, Morelande MR, Särkkä S (2015) Posterior linearization filter: Principles and implementation using sigma points. IEEE Transactions on Signal Processing 63(20):5561–5573
- Golub GH, Welsch JH (1969) Calculation of Gauss quadrature rules. Mathematics of Computation 23(106):221–230
- Grewal MS, Andrews AP (2001) Kalman Filtering: Theory and Practice Using MATLAB. John Wiley & Sons, Inc.
- Hartikainen J, Särkkä S (2010) Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In: IEEE International Workshop on Machine Learning for Signal Processing (MLSP), pp 379–384
- Hennig P, Hauberg S (2014) Probabilistic solutions to differential equations and their application to Riemannian statistics. In: Proc. of the 17th int. Conf. on Artificial Intelligence and Statistics (AISTATS), JMLR, W&CP, vol 33
- Hennig P, Osborne M, Girolami M (2015) Probabilistic numerics and uncertainty in computations. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 471(2179)
- Hochbruck M, Ostermann A, Schweitzer J (2009) Exponential Rosenbrock-type methods. SIAM Journal on Numerical Analysis 47(1):786–803
- Kennedy M, O’Hagan A (2002) Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B 63(3)
- Kersting H, Hennig P (2016) Active uncertainty calibration in Bayesian ODE solvers. Uncertainty in Artificial Intelligence (UAI)
- Kersting H, Sullivan T, Hennig P (2018) Convergence rates of Gaussian ODE filters. arXiv:180709737 [mathNA]
- Kimeldorf G, Wahba G (1970) A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. Ann Math Statist 41(2)
- Lie H, Stuart A, Sullivan T (2017) Strong convergence rates of probabilistic integrators for ordinary differential equations. arXiv:170303680 [mathNA]
- Loscalzo F, Talbot T (1967) Spline function approximations for solutions of ordinary differential equations. SIAM Journal on Numerical Analysis 4(3)
- McNamee J, Stenger F (1967) Construction of fully symmetric numerical integration formulas. Numerische Mathematik 10(4):327–344
- Øksendal B (2003) Stochastic Differential Equations: An Introduction with Applications, 5th edn. Springer
- Paul S, Chatzilygeroudis K, Ciosek K, Mouret JB, Osborne MA, Whiteson S (2018) Alternating optimisation and quadrature for robust control. In: AAAI Conference on Artificial Intelligence

- Prüher J, Šimandl M (2015) Bayesian quadrature in nonlinear filtering. In: 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO), vol 01, pp 380–387
- Rasmussen C, Williams C (2006) Gaussian Processes for Machine Learning. MIT
- Ritter K (2000) Average-Case Analysis of Numerical Problems. Springer
- Särkkä S (2013) Bayesian Filtering and Smoothing. Institute of Mathematical Statistics Textbooks, Cambridge University Press
- Schober M, Duvenaud D, Hennig P (2014) Probabilistic ODE solvers with Runge–Kutta means. In: Advances in Neural Information Processing Systems (NIPS)
- Schober M, Särkkä S, Hennig P (2018) A probabilistic model for the numerical solution of initial value problems. *Statistics and Computing*
- Skilling J (1991) Bayesian solutions of ordinary differential equations. *Maximum Entropy and Bayesian Methods*, Seattle
- Teymur O, Zygalakis K, Calderhead B (2016) Probabilistic linear multistep methods. In: Advances in Neural Information Processing Systems (NIPS)
- Teymur O, Lie HC, Sullivan T, Calderhead B (2018) Implicit probabilistic integrators for ODEs. arXiv preprint arXiv:180507970
- Tronarp F, Garcia-Fernandez AF, Särkkä S (2018) Iterative filtering and smoothing in non-linear and non-Gaussian systems using conditional moments. *IEEE Signal Processing Letters* 25(3):408–412, DOI 10.1109/LSP.2018.2794767
- Wang J, Cockayne J, Oates C (2018) On the Bayesian solution of differential equations. arXiv:170303680 [mathNA]
- Zhang J, Mokhtari A, Sra S, Jadbabaie A (2018) Direct Runge–Kutta discretization achieves acceleration. In: Advances in Neural Information Processing Systems (NIPS)