Calibrated Adaptive Probabilistic ODE Solvers

Nathanael Bosch[1] **Philipp Hennig**[1,2] **Filip Tronarp**[1]
[1]University of Tübingen
[2]Max Planck Institute for Intelligent Systems, Tübingen, Germany
{nathanael.bosch, philipp.hennig, filip.tronarp}@uni-tuebingen.de

Abstract

Probabilistic solvers for ordinary differential equations assign a posterior measure to the solution of an initial value problem. The joint covariance of this distribution provides an estimate of the (global) approximation error. The contraction rate of this error estimate as a function of the solver's step size identifies it as a well-calibrated worst-case error, but its explicit numerical value for a certain step size is not automatically a good estimate of the explicit error. Addressing this issue, we introduce, discuss, and assess several probabilistically motivated ways to calibrate the uncertainty estimate. Numerical experiments demonstrate that these calibration methods interact efficiently with adaptive step-size selection, resulting in descriptive, and efficiently computable posteriors. We demonstrate the efficiency of the methodology by benchmarking against the classic, widely used Dormand–Prince 4/5 Runge–Kutta method.

1 INTRODUCTION

Ordinary differential equations (ODEs) arise in almost all areas of science and engineering. In the field of machine learning, recent work on normalizing flows (Rezende and Mohamed, 2015) and neural ODEs (Chen et al., 2018) lead to a particular surge of interest. In this paper we consider initial value problems (IVPs), defined by an ODE

$$\dot{y}(t) = f\left(y(t), t\right), \qquad \forall t \in [t_0, T], \tag{1}$$

with vector field $f : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ and initial value $y(t_0) = y_0 \in \mathbb{R}^d$.

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

Limited by finite computational resources, the numerical solution of an IVP is inevitably only an approximation. Though, most classic numerical solvers do not return an estimate of their own numerical error, leaving it to the practitioner to evaluate the reliability of the result. The field of *probabilistic numerics* (PN) (Hennig et al., 2015; Oates and Sullivan, 2019) seeks to overcome this ignorance of numerical uncertainty. By treating numerical algorithms as problems of statistical inference, the numerical error can be quantified probabilistically.

One particular class of probabilstic numerical solvers for ODEs treats IVPs as a Gauss–Markov regression problem (Tronarp et al., 2019, 2020), the solution of which can be efficiently approximated with Bayesian filtering and smoothing (Särkkä, 2013). These so-called ODE filters relate to classic multistep methods (Schober et al., 2018), and have been shown to converge to the true solution of the IVP with high polynomial rates while providing (asymptotically) well-calibrated confidence intervals (Kersting et al., 2020). In practice, however, there remain two gaps: First, efficient implementation of ODE solvers require adaptive step-size selection, which has not received much attention in the past; second, the calibration of the posterior uncertainty estimates depends on the choice of specific diffusion hyperparameters. Both are addressed in this paper.

The contributions of this paper are the following: We introduce and discuss uncertainty calibration methods for models with both constant and time-varying diffusion, and extend existing approaches with multivariate parameter estimates. After calibration, the probabilistic observation model provides an objective for local error control and adaptive step-size selection, enabling the solvers to make efficient use of their computational budget. The resulting probabilistic numerical ODE solvers are evaluated and compared for a large range of configurations and tolerance levels, demonstrating descriptive posteriors and computational efficiency comparable to the classic Dormand–Prince 4/5 Runge–Kutta method.

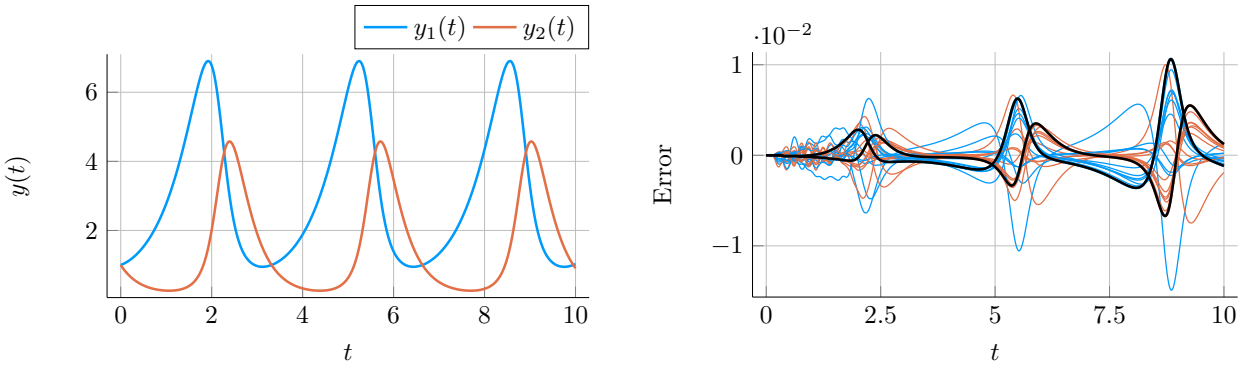

Figure 1: *Probabilistic solution of the Lotka-Volterra equations.* Left: Posterior mean returned by the probabilistic solver. For the chosen tolerance levels the returned uncertainties are too small to be visually separated from the mean. Right: True error trajectory (black) and sampled error trajectories (colored). Both the true solution and the samples exhibit similar patterns, indicating a well-structured and calibrated uncertainty estimate of the provided posterior distribution.

2 PROBABILISTIC ODE SOLVERS

In this paper, the solution of the IVP is posed as a Bayesian inference problem (Cockayne et al., 2019). By considering Gauss–Markov priors on the state-space, the problem is reduced to a case of Bayesian state estimation (Tronarp et al., 2019). In the following, we introduce the state estimation problem and describe the approximate inference procedure.

2.1 ODE Solutions as State Estimation

A priori, we model the solution together with $q \in \mathbb{N}$ of its derivatives as a Gauss–Markov process $X(t) = \left[\left(X^{(0)}(t)\right)^\intercal, \left(X^{(1)}(t)\right)^\intercal, \ldots, \left(X^{(q)}(t)\right)^\intercal\right]^\intercal$, where $X^{(i)}(t)$ models the i-th derivative $y^{(i)}(t)$. More precisely, $X(t)$ is the q-times integrated Wiener process (IWP), which solves the stochastic differential equations

$$\mathrm{d}X^{(i)}(t) = X^{(i+1)}(t)\,\mathrm{d}t, \quad i = 1, \ldots, q-1 \tag{2a}$$

$$\mathrm{d}X^{(q)}(t) = \Gamma^{1/2}\,\mathrm{d}B(t), \tag{2b}$$

$$X(t_0) \sim \mathcal{N}(\mu_0, \Sigma_0), \tag{2c}$$

where $\Gamma^{1/2}$ is the symmetric square root of some positive semi-definite matrix $\Gamma \in \mathbb{R}^{d\times d}$.

The continuous-time model can be described by transition densities (Särkkä and Solin, 2019, Section 6.2)

$$X(t+h)\,|\,X(t) \sim \mathcal{N}\left(A(h)X(t), Q(h)\right). \tag{3}$$

For the IWP prior, $A(h) \in \mathbb{R}^{d(q+1)\times d(q+1)}$ and $Q(h) \in \mathbb{R}^{d(q+1)\times d(q+1)}$ are of the form

$$A(h) = \breve{A}(h) \otimes I_d, \tag{4a}$$

$$Q(h) = \breve{Q}(h) \otimes \Gamma, \tag{4b}$$

with $\breve{A}(h), \breve{Q}(h)$ given by Kersting et al. (2020, Appendix A)

$$\breve{A}_{ij}(h) = \mathbb{I}_{i\le j}\frac{h^{j-1}}{(j-i)!}, \tag{5a}$$

$$\breve{Q}_{ij}(h) = \frac{h^{2q+1-i-j}}{(2q+1-i-j)(q-i)!(q-j)!}. \tag{5b}$$

To relate the prior to the solution of the IVP, define the measurement process

$$Z(t) = X^{(1)}(t) - f\left(X^{(0)}(t)\right). \tag{6}$$

The probabilistic numerical solution of the ODE is computed by conditioning $X(t)$ on the event that the realisation $z(t)$ of $Z(t)$ is zero on the grid $\{t_n\}_{n=1}^N$ (Tronarp et al., 2019)

$$z_n := z(t_n) = 0, \quad n = 1, \ldots, N. \tag{7}$$

The resulting inference problem of computing

$$p\left(X(t)\,|\,\{z_n\}_{n=1}^N\right) \tag{8}$$

is, for non-linear vector fields f, known as a non-linear Gauss–Markov regression problem and is in general intractable, but it is possible to efficiently compute approximations (Särkkä, 2013).

2.2 Approximate Gaussian Inference

We consider approximate Bayesian inference based on linearization by Taylor-series expansion, known as the extended Kalman filter (EKF) in statistical signal processing (Särkkä, 2013, Section 5.2). By linearizing the

measurement likelihood we can efficiently and iteratively compute approximations

$$p\left(X(t_n) \,|\, \{z_i\}_{i=1}^{n-1}\right) \approx \mathcal{N}\left(\mu_n^P, \Sigma_n^P\right), \tag{9a}$$
$$p\left(X(t_n) \,|\, \{z_i\}_{i=1}^{n}\right) \approx \mathcal{N}\left(\mu_n^F, \Sigma_n^F\right), \tag{9b}$$
$$p\left(z_n \,|\, \{z_i\}_{i=1}^{n-1}\right) \approx \mathcal{N}\left(\hat{z}_n, S_n\right), \tag{9c}$$

through the following *prediction* and *update* steps (Särkkä, 2013, Section 5.2):

Prediction:

$$\mu_n^P = A(h_{n-1})\mu_{n-1}^F, \tag{10a}$$
$$\Sigma_n^P = A(h_{n-1})\Sigma_{n-1}^F A(h_{n-1})^\intercal + Q(h_{n-1}). \tag{10b}$$

Update:

$$\hat{z}_n = E_1\mu_n^P - f\left(E_0\mu_n^P, t_n\right), \tag{11a}$$
$$S_n = H_n\Sigma_n^P H_n^\intercal, \tag{11b}$$
$$K_n = \Sigma_n^P H_n^\intercal S_n^{-1}, \tag{11c}$$
$$\mu_n^F = \mu_n^P + K_n(z_n - \hat{z}_n), \tag{11d}$$
$$\Sigma_n^F = \Sigma_n^P - K_n S_n K_n^\intercal, \tag{11e}$$

where H_n can be either $H_n := E_1$ for a zeroth order approximation, or $H_n := E_1 - J_f(E_0\mu_n, t_n)E_0$ for a first order approximation of the vector field f, with $E_0 := e_0^\intercal \otimes I$, $E_1 := e_1^\intercal \otimes I$.

The zeroth and first order linearizations correspond to the updates by Schober et al. (2018) and Tronarp et al. (2019), respectively. In the sequel, we refer to the algorithm with zeroth and first order linearization as EKF0 and EKF1, respectively.

Remark 1. *While most classic ODE solvers do not use the Jacobians of the vector field f, they play a central role in Rosenbrock methods (Rosenbrock, 1963; Hochbruck et al., 2008), a class of semi-implicit solvers for stiff ODEs (Hairer and Wanner, 1996, Chapter IV.7). In probabilistic solvers, the Jacobian was used in a probabilistic multistep method (Teymur et al., 2016) and, more recently, with extended Kalman filtering and smoothing (Tronarp et al., 2019, 2020).*

The Bayesian *filtering* posterior for $X(t)$ is conditioned only on the measurements obtained before and at the time step t, but does not include future measurements. Computing the (approximate) full marginal posterior $p\left(X(t_n) \,|\, \{z_n\}_{n=1}^N\right)$ can be done with Bayesian *smoothing*. The extended Rauch–Tung–Striebel smoother, also called the extended Kalman smoother (EKS), describes an algorithm to efficiently compute Gaussian approximations

$$p\left(X(t_n) \,|\, \{z_i\}_{i=1}^N\right) \approx \mathcal{N}\left(\mu_n^S, \Sigma_n^S\right) \tag{12}$$

with a backwards recursion, given by the *smoothing* step

$$G_n = \Sigma_n^F A(h_n)(\Sigma_{n+1}^P)^{-1}, \tag{13a}$$
$$\mu_n^S = \mu_n^F + G_n(\mu_{n+1}^S - \mu_{n+1}^P), \tag{13b}$$
$$\Sigma_n^S = \Sigma_n^F + G_n(\Sigma_{n+1}^S - \Sigma_{n+1}^P)G_n^\intercal. \tag{13c}$$

See also Särkkä (2013, Chapter 9). The approximate posterior for off-the-grid time steps $t \in [t_0, T]$, relating to *dense output* in classic numerical solvers (Hairer et al., 1993, Chapter II.6), can be straight-forwardly computed by using the Gauss–Markov property. In a similar manner as above, we refer to the resulting algorithms with linearization of order zero and one as EKS0 and EKS1, respectively.

The question remains how to set the initial state $X_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$. This problem can also be observed in classic multistep methods: In addition to the multistep formula, they specify a starting procedure using, for example, Taylor series expansion (Bashforth and Adams, 1883), one-step methods (Hairer et al., 1993, Chapter III.1), or other iterative procedures (Nordsieck, 1962). In a similar effort, Schober et al. (2018) describe an initialization for a probabilistic ODE solver via Runge-Kutta methods. Since the probabilistic formulation enables us to explicitly quantify uncertainty over the initial values, it is also possible to set the initial state to a zero mean and unit variance Gaussian distribution and to condition on the correct initial values $X^{(0)}(t_0) = y_0$ and $X^{(1)}(t_0) = f(y_0, t_0)$ (Tronarp et al., 2019, 2020). For our experiments, where orders $q \leq 5$ are used, we found it feasible to compute all derivatives of the correct initial value via automatic differentiation, and we explicitly set $\mu_0 = (y(t_0), \dot{y}(t_0), ..., y^{(q)}(t_0))$ and zero covariance.

Remark 2. *The exact initial derivatives can be computed efficiently with Tylor-mode automatic differentiation (Griewank and Walther, 2000; Bettencourt et al., 2019). A more extensive description of an initialization procedure, together with further considerations for a numerically stable implementation, is provided by Krämer and Hennig (2020).*

3 UNCERTAINTY CALIBRATION

The probabilistic solver with IWP prior presented in Section 2.1 contains a free parameter Γ, which is of particular importance for the posterior uncertainty, as it determines the gain of the Wiener process entering the system in Eq. (2). In this section, we present different diffusion models and discuss approaches for estimating this parameter and thereby calibrating uncertainties.

3.1 Time-Fixed Diffusion Model

A common approach for Bayesian model selection and parameter estimation is to maximize the marginal likelihood, or *evidence*, of the observed data $z_{1:n}$, given by the prediction error decomposition (Schweppe, 1965)

$$p(z_{1:n}) = p(z_1) \prod_{i=2}^{n} p(z_i \mid z_{1:i-1}). \tag{14}$$

For affine vector fields, the Kalman filter computes the marginals $p(z_i \mid z_{1:i-1})$ exactly, but for non-affine vector fields we solve the Bayesian filtering problem only approximately. Nevertheless, it is a natural choice to approximate the marginal likelihoods in the same way as the filtering solution is approximated, i.e. with extended Kalman filtering, as

$$p(z_{1:n}) \approx \prod_{i=1}^{n} \mathcal{N}\left(z_i; \hat{z}_i, S_i\right), \tag{15a}$$

$$\hat{z}_i = E_1 \mu_i^P - f\left(E_0 \mu_i^P, t_i\right), \tag{15b}$$

$$S_i = H_i \Sigma_i^P H_i^\intercal. \tag{15c}$$

Maximizing Eq. (15a) is referred to as *quasi maximum likelihood estimation* in signal processing (Lindström et al., 2018).

For the case of scalar matrices $\Gamma = \sigma^2 I_d$, Tronarp et al. (2019, Proposition 4) provide a (quasi) maximum-likelihood estimate (quasi-MLE) of σ^2, denoted by $\hat{\sigma}^2_N$. Assuming an initial covariance of the form $\Sigma_0 = \sigma^2 \breve{\Sigma}_0$, $\hat{\sigma}^2_N$ is given by

$$\hat{\sigma}^2_N = \frac{1}{Nd} \sum_{n=1}^{N} (z_n - \hat{z}_n)^\intercal S_n^{-1} (z_n - \hat{z}_n). \tag{16}$$

This estimation can be performed on-line in order to provide calibrated uncertainty estimates during the solve, which are required for step-size adaptation.

The IWP prior with scalar diffusion $\Gamma = \sigma^2 I_d$ describes the same model for each dimension. Furthermore, as the measurement matrix H_n associated with the EKF0 does not depend on the vector field, the estimated uncertainties of each dimension will be the same. To fix this shortcoming of the EKF0 we propose a model with diagonal $\Gamma = \operatorname{diag}(\sigma_1^2, \ldots, \sigma_d^2)$. Its quasi-MLE is provided in Proposition 1 below.

Proposition 1. *Let $\Gamma = \operatorname{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ and $\Sigma_0 = \breve{\Sigma}_0 \otimes \Gamma$. Then the prediction and filtering covariances computed by the EKF0 with IWP prior are of the form $\Sigma_n^P = \breve{\Sigma}_n^P \otimes \Gamma$, $\Sigma_n^F = \breve{\Sigma}_n^F \otimes \Gamma$, and the approximated measurement covariances are given by $S_n = \breve{s}_n \cdot \Gamma$, where $\breve{s}_n$ is $\breve{s}_n := e_2^\intercal \breve{\Sigma}_n^P e_2$. The quasi maximum-likelihood estimate of Γ, denoted by $\hat{\Gamma}$, is diagonal and given by*

$$\hat{\Gamma}_{ii} = \frac{1}{N} \sum_{n=1}^{N} \frac{(\hat{z}_n)_i^2}{\breve{s}_n}, \qquad i \in \{1, \ldots, d\}. \tag{17}$$

The proof follows the idea of Tronarp et al. (2019, Proposition 4). A more detailed derivation can be found in Appendix A.

3.2 Time-Varying Diffusion Model

To allow for greater flexibility, Schober et al. (2018) propose a model in which $\Gamma = \Gamma_n$ is allowed to vary for different integration steps t_n. In such a model, all measurements $\{z_i\}_{i=1}^{n-1}$ taken before time t_n are independent of the parameter Γ_n. We obtain

$$\arg\max_{\Gamma_n} p\left(z_{1:n}\right) = \arg\max_{\Gamma_n} p\left(z_n \mid z_{1:n}\right) \tag{18a}$$

$$\approx \arg\max_{\Gamma_n} \mathcal{N}\left(z_n; \hat{z}_n, S_n\right), \tag{18b}$$

with $S_n = H_n \left[A_n \Sigma_{n-1}^F A_n^\intercal + (\breve{Q}_n \otimes \Gamma_n)\right] H_n^\intercal$.

To approximately estimate Γ_n, Schober et al. (2018) propose an estimation based on "local" errors, a common procedure for error control and step-size adaptation in classic numerical methods (Hairer et al., 1993, Chapter II.4). Assuming an error-free predicted solution μ_{n-1}^F at time t_{n-1}, that is, $\Sigma_{n-1}^F = 0$, yields

$$S_n = H_n (\breve{Q}_n \otimes \Gamma_n) H_n^\intercal. \tag{19}$$

For scalar matrices $\Gamma_n = \sigma_n^2 I_d$ this implies

$$S_n = \sigma_n^2 \cdot H_n (\breve{Q}_n \otimes I_d) H_n^\intercal. \tag{20}$$

Computing the quasi-MLE by solving Eq. (18) yields the parameter estimate by Schober et al. (2018):

$$\hat{\sigma}_n^2 = \frac{1}{d} (z_n - \hat{z}_n)^\intercal \left(H_n (\breve{Q}_n \otimes I_d) H_n^\intercal\right)^{-1} (z_n - \hat{z}_n). \tag{21}$$

As for the fixed diffusion model, we can improve the expressiveness of the EKF0 by considering a multivariate model with diagonal $\Gamma_n = \operatorname{diag}(\sigma_{n1}^2, \ldots, \sigma_{nd}^2)$. With the local error based estimation, and using $H_n = e_1 \otimes I_d$, we obtain

$$S_n = (\breve{Q}_n)_{11} \cdot \Gamma_n. \tag{22}$$

With a covariance of this form, Eq. (18) can be solved and we obtain the quasi-MLE

$$(\hat{\Gamma}_n)_{ii} = (z_n - \hat{z}_n)_i^2 / (\breve{Q}_n)_{11}, \qquad i \in \{1, \ldots, d\}, \tag{23}$$

as parameter estimate for models with the time-varying, diagonal diffusion.

4 STEP-SIZE ADAPTATION

While the algorithm described in Sections 2 and 3 is able to compute calibrated posterior distributions over

solutions to any IVP, it still lacks a common tool to make efficient use of its computations: *step-size adaptation*. Indeed, most modern, non-probabilistic ODE solvers perform local error control with adaptive step-size selection, allowing them to compute the result up to a desired precision while avoiding unnecessary computational work. In the following, we first review error estimation and step-size control in classic numerical solvers, then provide a principled objective for error control of probabilistic ODE solvers and describe the full step-size adaptation algorithm.

4.1 Error Control in Classic Solvers

An important aspect of numerical analysis is to monitor and control the error of a method. Commonly, a distinction is made between two kinds of errors: The *local* error describes the error that the algorithm introduces after a single step, whereas the *global* error is the cumulative error of the computed solution caused by multiple iterations. The global error is typically not of practical interest for error monitoring and control (Hairer et al., 1993, Chapter II.4), and instead the local error is estimated, for example through Richardson extrapolation (Hairer et al., 1993, Theorem 4.1), or more commonly via embedded Runge-Kutta methods (Hairer et al., 1993, Chapter II.4) or the Milne device (Byrne and Hindmarsh, 1975). These estimates are then used in the step-size control algorithm, which ensures that the chosen step sizes are sufficiently small to yield the desired precision of the computed result, while being sufficiently large to avoid unnecessary computational work. Common control algorithms for step-size selection include proportional control (Hairer et al., 1993, Chapter II.4) and proportional-integral (PI) control (Gustafsson et al., 1988).

4.2 Error Control in Probabilistic Solvers

In Gaussian filtering, the natural object to consider for error estimation and control are the residuals $(z_n - \hat{z}_n)$. Schober et al. (2018) show how to use this quantity for both uncertainty calibration (as presented in Section 3.2) and local error control. We generalize their error control objective to be applicable to all presented algorithms and uncertainty calibration methods.

After calibration, the extended Kalman filtering algorithm approximates (see Eq. (11))

$$p(z_n \,|\, z_{1:n-1}) \approx \mathcal{N}\left(z_n; \hat{z}_n, S_n\right), \tag{24}$$

with

$$\hat{z}_n = E_1 \mu_n^P - f\left(E_0 \mu_n^P, t_n\right), \tag{25a}$$

$$S_n = H_n \left(A_n \Sigma_{n-1}^F A_n^\intercal + \left(\breve{Q}_n \otimes \hat{\Gamma}_n\right)\right) H_n^\intercal, \tag{25b}$$

where H_n can correspond to either the zeroth or the first order linearization, and $\hat{\Gamma}_n$ has been estimated through one of the approaches of Section 3.

For step-size adaptation we want to control *local* errors, and therefore assume an error-free solution estimate at time t_{n-1}. With $\Sigma_{n-1}^F = 0$ we obtain the approximation

$$p((z_n - \hat{z}_n) \,|\, z_{1:n-1}) \approx \mathcal{N}\left(z_n - \hat{z}_n; 0, H_n \left(\breve{Q}_n \otimes \hat{\Gamma}_n\right) H_n^\intercal\right). \tag{26}$$

Finally, we define the objective for local error control $D_n \in \mathbb{R}^d$ as the (local) standard deviations of the residual vector $(z_n - \hat{z}_n)$, given by

$$(D_n)_i := \left(H_n \left(\breve{Q}_n \otimes \hat{\Gamma}_n\right) H_n^\intercal\right)_{ii}^{1/2}, \quad i \in \{1, \ldots, d\}. \tag{27}$$

For the EKF0 algorithm and time-varying, scalar $\Gamma = \sigma^2 I_d$, we recover the expected error used by Schober et al. (2018) for step-size control (see also Byrne and Hindmarsh (1975)), but the general formulation in Eq. (27) can also be used with on-line quasi-MLE for time-fixed Γ (Section 3.1) and in combination with the EKF1.

4.3 Step-Size Selection

Following Hairer et al. (1993, Chapter II.4), the step-size controller aims to select step sizes, as large as possible, but while satisfying componentwise, for $i \in \{1, \ldots, d\}$,

$$(D_n)_i \leq \varepsilon_i, \quad \varepsilon_i := \tau_{\text{abs}} + \tau_{\text{rel}} \cdot \max\left(|(\hat{y}_{n-1})_i|, |(\hat{y}_n)_i|\right), \tag{28}$$

where τ_{abs} and τ_{rel} are the prescribed absolute and relative tolerances, respectively, and $\hat{y}_{n-1} := E_0 \mu_{n-1}^F$, $\hat{y}_n := E_0 \mu_n^F$ are solution estimates of the numerical solver. To do so, we define the following measure of error as control objective,

$$E := \sqrt{\frac{1}{d} \sum_{i=1}^{d} \left(\frac{(D_n)_i}{\varepsilon_i}\right)^2}. \tag{29}$$

The proportional control algorithm compares the control objective E to 1 to find the optimal step size. If $E \leq 1$ holds, the computed step is accepted and the integration continues. Otherwise, the step is rejected as too inaccurate and is repeated. In both cases, a new step size which will likely satisfy Eq. (28) is computed as

$$h_{\text{new}} = h \cdot \rho \left(\frac{1}{E}\right)^{\frac{1}{q+1}}, \tag{30}$$

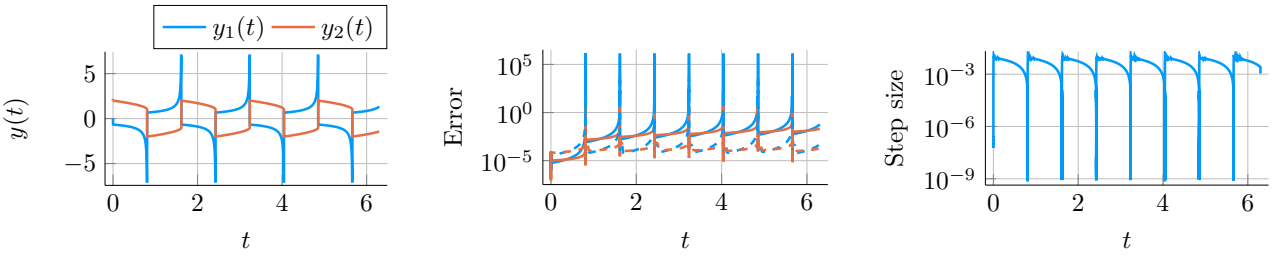

Figure 2: *Probabilistic solution with step-size adaptation of the Van der Pol equations.* Left: Mean of the posterior distribution over solutions, showing only values in the $(-7, 7)$ interval. Middle: Absolute errors (solid lines), and standard deviations of the posterior marginals as error estimates (dashed lines), shown in log-scale. Right: Step sizes of accepted steps throughout the solve. During the stiff phases, the step sizes get drastically decreased by the step-size controller.

making use of the local convergence rate of $q + 1$, as used by Schober et al. (2018) and shown by Kersting et al. (2020). The parameter $\rho \in (0, 1]$ is a safety factor to increase the probability that the next step will be acceptable, and we additionally limit the rate of change $\eta_{\min} \leq h_{n+1}/h_n \leq \eta_{\max}$ (Hairer et al., 1993). In our experiments, we set $\rho := 0.9$, $\eta_{min} := 0.2$, $\eta_{max} := 10$.

The formulation of the local error control objective in Eq. (27) also lends itself to other control algorithms. Notably, PI control (Gustafsson et al., 1988) can be an interesting alternative to proportional control when applied to mildly stiff problems, and has been successfully applied for the related class of Nordsieck methods by Bras et al. (2013).

5 RELATED WORK

Our contribution fits in the formulation of IVPs as problems of Bayesian state estimation (Tronarp et al., 2019, 2020). By using Gaussian filtering methods, these solvers are able to efficiently compute posterior distributions over solutions (Kersting and Hennig, 2016), which converge to the true solution at high polynomial rates (Kersting et al., 2020). In practice, the calibration of the posterior uncertainties depends on specific model hyperparameters. This paper reviews and extends previously proposed global and local calibration methods (Tronarp et al., 2019; Schober et al., 2018). The presented step-size controller builds on the algorithm suggested by Schober et al. (2018).

A different line of work on probabilistic numerical solvers for ODEs aims to represent the distribution over solution with a set of sample paths (Conrad et al., 2017; Abdulle and Garegnani, 2020; Lie et al., 2019; Teymur et al., 2018, 2016; Chkrebtii et al., 2016; Tronarp et al., 2019). While these methods are able to capture arbitrary, non-Gaussian distributions, they come at an increased computational cost.

6 EXPERIMENTS

To evaluate the presented methodology, we provide three sets of experiments. First, we highlight the practical necessity of step-size adaptation by solving a stiff version of the Van der Pol model. Next, we compare the different uncertainty calibration methods presented in Section 3. Finally, we assess the practical performance of the probabilistic ODE solvers by comparing to a classic Runge-Kutta 4/5 method. The code for the implementation and experiments is publicly available on gihub[1].

6.1 Stiff Van der Pol

The Van der Pol model (van der Pol, 1926) describes a non-conservative oscillator with non-linear damping, and can be written in the two-dimensional form:

$$\begin{aligned} \dot{y_1} &= y_2, \\ \dot{y_2} &= \mu \left((1 - y_1^2) y_2 - y_1 \right), \end{aligned} \tag{31}$$

with a positive stiffness constant $\mu > 0$.

To highlight the importance of step-size adaptation, and to demonstrate the A-stability of the EKS1 (Tronarp et al., 2020), we consider a very stiff version of the Van der Pol model and set $\mu = 10^6$. We solve the IVP on the time interval $[0, 6.3]$ with initial value $y(t_0) = [0, \sqrt{3}]^\intercal$ using the EKS1 algorithm with an IWP3 prior, for absolute and relative tolerance specified as 10^{-6} and 10^{-3}, respectively. We were not able to solve this same IVP with the EKS0 (which does not possess this stability property). The reference solution has been computed with the A-stable 5th order implicit Runge-Kutta method Radau IIA (Hairer and Wanner, 1996), implemented as `RadauIIA5` in the Julia DifferentialEquations.jl suite (Rackauckas and Nie, 2017), for absolute and relative tolerances set to 10^{-14}.

[1] `https://github.com/nathanaelbosch/capos`

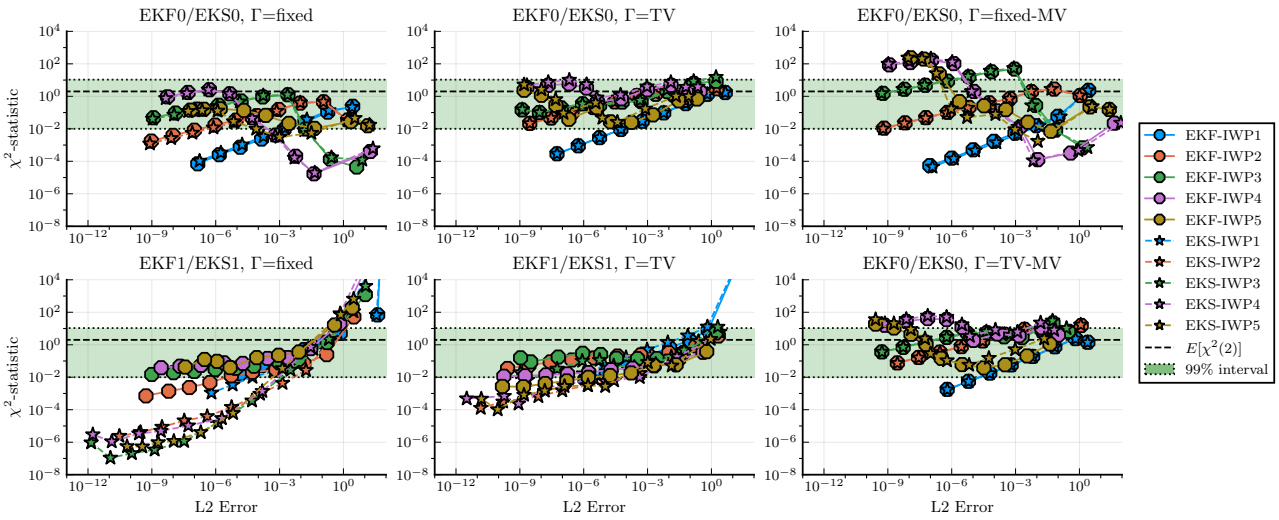

Figure 3: *Uncertainty calibration across configurations.* In each subfigure, a specific combination of filtering algorithm (EKF0/EKS0 or EKS1/EKF1) and calibration method is evaluated, the latter including fixed and time-varying (TV) diffusion models, as well as their multivariate versions (fixed-MV, TV-MV). A well-calibrated solver should provide χ^2-statistics inside the 99% credible interval (green).

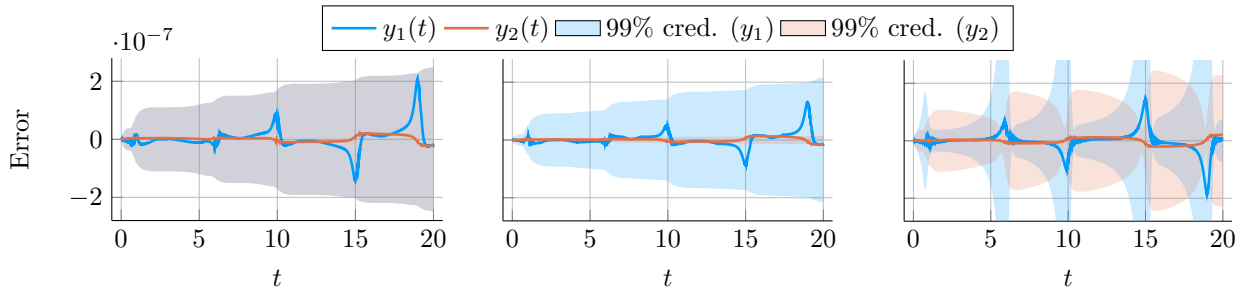

Figure 4: *Qualitative comparison of the different time-varying uncertainty models.* The EKS0 with scalar diffusion (left) estimates a single credible band shared accross for both dimensions, but the multivariate model (middle) is able to attribute different uncertainties to each dimension. By including information on the derivatives of the ODE, the EKS1 with scalar diffusion (right) returns structured and dynamic uncertainty estimate.

Figure 2 shows the results, including the solution posterior, errors and error estimates, and the step sizes during the solve. The posterior mean returned by the solver achieved a final error of $\|\hat{y}(T) - y^*(T)\| = 6.17 \times 10^{-2}$, where y^* denotes the reference solution. The step sizes shown in Fig. 2 (right) change drastically during the solve, and decrease from values $h \sim 10^{-2}$ down to $h < 10^{-8}$ during the stiff phases. If one were to solve this IVP without step-size control, that is, with EKS1 and an IWP3 prior but with fixed steps of size 10^{-8}, one would perform 6.3×10^8 solver iterations. Compared to the ~ 1 second runtime the adaptive step method required for its 23824 iterations (out of which 6977 steps were rejected), the fixed-step solver would require more than 7 hours. This demonstrates the importance of adaptive step-size control for efficient use of computational resources.

6.2 Comparison of Calibration Methods

We evaluate the various algorithms and calibration methods on the FitzHugh-Nagumo model, given by the ODE

$$\begin{aligned} \dot{y}_1 &= c\left(y_1 - \frac{y_1^3}{3} + y_2\right), \\ \dot{y}_2 &= -\frac{1}{c}\left(y_1 - a - b y_2\right), \end{aligned} \tag{32}$$

with parameters ($a = 0.2$, $b = 0.2$, $c = 3.0$), initial value $y(0) = [-1, 1]^\intercal$, and time span $[0, 20]$.

In the first part of this experiment, the uncertainty calibration is evaluated across a large range of solver configurations and tolerances ($\tau_{\text{abs}} = 10^{-4}, \ldots, 10^{-13}$, $\tau_{\text{rel}} = 10^{-1}, \ldots, 10^{-10}$). To assess the quality of the uncertainty calibration, we use the χ^2-statistics (Bar-

Shalom et al., 2004) defined by

$$\chi^2 = \frac{1}{N}\sum_{i=1}^{N} r(t_i)^\intercal \operatorname{Cov}(y(t_i))^{-1} r(t_i), \qquad (33)$$

where $r(t_i) := (y^*(t_i) - \mathbb{E}[y(t_i)])$ are the residuals and $\mathbb{E}[y(t_i)]$ and $\operatorname{Cov}(y(t_i))$ are computed on the posterior distribution returned by the probabilistic solver. A well calibrated model achieves $\chi^2 \approx d$. If $\chi^2 < d$ or $\chi^2 > d$ we refer to the solution as *underconfident* or *overconfident*, respectively.

Figure 3 visualizes the full comparison of the uncertainty calibration methods and suggests some empirical findings. For most configurations the time-varying uncertainty models seem to be better calibrated than the fixed models. For a fixed choice of algorithm (e.g. EKF0) and order (e.g. IWP5) their calibration varies less across the different tolerance levels. The multi-variate models seem to not have a large impact on the χ^2-statistics, both for the fixed and time-varying models. The first-order linearization approaches EKF1/EKS1 tend to become underconfident, but achieve the lowest errors.

To complement this summarized evaluation based on the χ^2 statistics, we visualize the qualitative behaviour of the different time-varying uncertainty models in Fig. 4. To be comparable, all models share an IWP3 prior and tolerances $\tau_{\text{abs}} = 10^{-10}$, $\tau_{\text{rel}} = 10^{-7}$. The scalar, time-varying (TV) approach attributes the same credible bands to both dimensions (shown in grey), whereas the multivariate, time-varying (TV-MV) model is able to lift this restriction and estimates a large uncertainty for the first dimension (blue) and barely visible uncertainties for the second dimension (orange). However, only the first order linearization of the EKS1 seems to properly describe the structural properties of the true solution in its posterior estimate.

For experiments on additional problems, including classic work-precision diagrams for all methods, see Appendix B.1

6.3 Comparison with Dormand–Prince 4/5

This experiment assesses the performance of the developed methodology and compares the probabilistic solvers of 5th order to the classic, widely used Runge-Kutta 4/5 method by Dormand and Prince (1980), implemented as `DP5` in the Julia DifferentialEquations.jl suite (Rackauckas and Nie, 2017). The comparison was made on the Lotka-Volterra equations (Lotka, 1925; Volterra, 1928), which describe the dynamics of biological systems in which two species interact, one as a predator and the other as prey. The IVP is given by

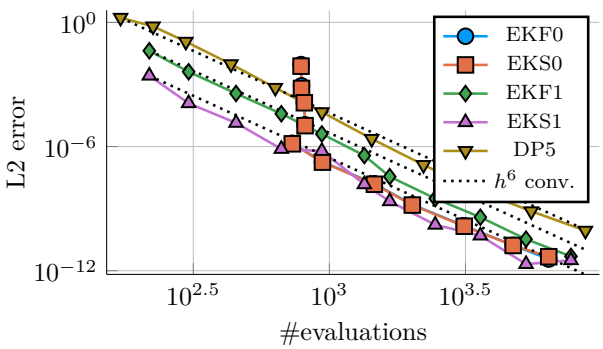

Figure 5: *Comparison to Dormand–Prince 4/5 (DP5).* All probabilistic ODE solvers use an IWP-5 prior and a scalar, time-varying diffusion model. The comparison is made on the Lotka-Volterra equations. The EKF0 and EKS0 performed similarly and are therefore difficult to separate visually in the figure.

the ODE

$$\begin{aligned} \dot{y_1} &= \alpha y_1 - \beta y_1 y_2, \\ \dot{y_2} &= -\gamma y_1 + \delta y_1 y_2, \end{aligned} \qquad (34)$$

with initial value $y(0) = [1, 1]^\intercal$ and parameters ($\alpha = 1.5$, $\beta = 1$, $\gamma = 3$, $\delta = 1$), on the time span $[0, 10]$.

Figure 5 shows the results in a work-precision diagram. We observe convergence rates of order 6 for all methods, one order higher than the expected global convergence rate of order 5 for the DP5 algorithm (Hairer et al., 1993) and for Gaussian ODE filters with IWP-5 prior (Kersting et al., 2020; Tronarp et al., 2020). It can also be seen that the EKF0 requires more function evaluations than expected for high-tolerance settings. Out of all compared methods, the EKS1 seems to require the least number of evaluations of the function and its Jacobian to achieve a specified error, matching the performance of the EKF0 and EKS0 while demonstrating a stable behaviour for high tolerances.

Similar work-precision diagrams for these methods on additional problems are provided in Appendix B.2.

7 CONCLUSION

In this paper, we introduced and discussed various models and methods for uncertainty calibration in Gaussian ODE filters, and presented parameter estimates for both fixed and time-varying, as well as scalar and multivariate diffusion models. The probabilistic observation model of these methods provides a calibrated objective for local error control, enabling the implementation of classic step-size selection algorithms.

The resulting, efficiently computable posteriors have

been empirically evaluated for a wide range of tolerance levels, demonstrating decent error calibration in particular for the time-varying diffusion models. Of all compared methods, the first-order linearization of the EKS1 seems to provide the most expressive posterior covariances, while also efficiently computing accurate solutions – requiring, in our benchmarks, less evaluations than the well-known Dormand–Prince 4/5 method to reach a specified tolerance level.

Acknowledgements

The authors gratefully acknowledge financial support by the German Federal Ministry of Education and Research (BMBF) through Project ADIMEM (FKZ 01IS18052B), and financial support by the European Research Council through ERC StG Action 757275 / PANAMA; the DFG Cluster of Excellence "Machine Learning - New Perspectives for Science", EXC 2064/1, project number 390727645; the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039A); and funds from the Ministry of Science, Research and Arts of the State of Baden-Württemberg. The authors also thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting N. Bosch.

The authors are grateful to Nicholas Krämer for many valuable discussions. They further thank Hans Kersting, Jonathan Wenger and Agustinus Kristiadi for helpful feedback on the manuscript.

References

Abdulle, A. and Garegnani, G. (2020). Random time step probabilistic methods for uncertainty quantification in chaotic and geometric numerical integration. *Statistics and Computing*, 30(4):907–932.

Bar-Shalom, Y., Li, X., and Kirubarajan, T. (2004). *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. Wiley.

Bashforth, F. and Adams, J. C. (1883). *An attempt to test the theories of capillary action by comparing the theoretical and measured forms of drops of fluid*. University Press.

Bettencourt, J., Johnson, M. J., and Duvenaud, D. (2019). Taylor-mode automatic differentiation for higher-order derivatives in jax.

Bras, M., Cardone, A., and D'Ambrosio, R. (2013). Implementation of explicit Nordsieck methods with inherent quadratic stability. *Mathematical Modelling and Analysis*, 18(2):289–307.

Byrne, G. D. and Hindmarsh, A. C. (1975). A polyalgorithm for the numerical solution of ordinary differential equations. *ACM Trans. Math. Softw.*

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

Chkrebtii, O. A., Campbell, D. A., Calderhead, B., and Girolami, M. A. (2016). Bayesian solution uncertainty quantification for differential equations. *Bayesian Anal.*, 11(4):1239–1267.

Cockayne, J., Oates, C., Sullivan, T., and Girolami, M. (2019). Bayesian probabilistic numerical methods. *SIAM Rev.*, 61:756–789.

Conrad, P. R., Girolami, M., Särkkä, S., Stuart, A., and Zygalakis, K. (2017). Statistical analysis of differential equations: introducing probability measures on numerical solutions. *Statistics and Computing*, 27(4):1065–1082.

Dormand, J. R. and Prince, P. J. (1980). A family of embedded Runge-Kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26.

Griewank, A. and Walther, A. (2000). Evaluating derivatives - principles and techniques of algorithmic differentiation, second edition. In *Frontiers in applied mathematics*.

Gustafsson, K., Lundh, M., and Söderlind, G. (1988). A PI stepsize control for the numerical solution of ordinary differential equations. *BIT Numerical Mathematics*, 28(2):270–287.

Hairer, E., Norsett, S., and Wanner, G. (1993). *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 8. Springer-Verlag.

Hairer, E. and Wanner, G. (1996). *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, volume 14. Springer-Verlag.

Hennig, P., Osborne, M. A., and 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., and Schweitzer, J. (2008). Exponential Rosenbrock-type methods. *SIAM J. Numer. Anal.*, 47(1):786–803.

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, page 309–318, Arlington, Virginia, USA. AUAI Press.

Kersting, H., Sullivan, T. J., and Hennig, P. (2020). Convergence rates of gaussian ode filters. *Statistics and Computing*, 30(6):1791–1816.

Krämer, N. and Hennig, P. (2020). Stable implementation of probabilistic ode solvers.

Lie, H. C., Stuart, A. M., and Sullivan, T. J. (2019). Strong convergence rates of probabilistic integrators for ordinary differential equations. *Statistics and Computing*, 29(6):1265–1283.

Lindström, E., Madsen, H., and Nielsen, J. (2018). *Statistics for Finance: Texts in Statistical Science*. Chapman and Hall/CRC.

Lotka, A. (1925). *Elements of Physical Biology*. Williams & Wilkins.

Nordsieck, A. (1962). On numerical integration of ordinary differential equations. *Mathematics of Computation*, 16:22–49.

Oates, C. J. and Sullivan, T. J. (2019). A modern retrospective on probabilistic numerics. *Statistics and Computing*, 29(6):1335–1351.

Rackauckas, C. and Nie, Q. (2017). DifferentialEquations.jl – a performant and feature-rich ecosystem for solving differential equations in julia. *Journal of Open Research Software*, 5(1).

Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. volume 37 of *Proceedings of Machine Learning Research*, pages 1530–1538, Lille, France. PMLR.

Rosenbrock, H. H. (1963). Some general implicit processes for the numerical solution of differential equations. *Comput. J.*, 5(4):329–330.

Särkkä, S. (2013). *Bayesian Filtering and Smoothing*, volume 3 of *Institute of Mathematical Statistics textbooks*. Cambridge University Press.

Schober, M., Särkkä, S., and Hennig, P. (2018). A probabilistic model for the numerical solution of initial value problems. *Statistics and Computing*.

Schweppe, F. (1965). Evaluation of likelihood functions for Gaussian signals. *IEEE transactions on Information Theory*, 11(1):61–70.

Särkkä, S. and Solin, A. (2019). *Applied Stochastic Differential Equations*. Institute of Mathematical Statistics Textbooks. Cambridge University Press.

Teymur, O., Lie, H. C., Sullivan, T., and Calderhead, B. (2018). Implicit probabilistic integrators for odes. In *Advances in Neural Information Processing Systems 31*, pages 7244–7253. Curran Associates, Inc.

Teymur, O., Zygalakis, K., and Calderhead, B. (2016). Probabilistic linear multistep methods. In *Advances in Neural Information Processing Systems 29*, pages 4321–4328. Curran Associates, Inc.

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*, 29(6):1297–1315.

Tronarp, F., Sarkka, S., and Hennig, P. (2020). Bayesian ode solvers: the maximum a posteriori estimate. *CoRR*.

van der Pol, B. (1926). On "relaxation-oscillations". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):978–992.

Volterra, V. (1928). Variations and Fluctuations of the Number of Individuals in Animal Species living together. *ICES Journal of Marine Science*, 3(1):3–51.