Bayesian Uncertainty Quantification for Differential Equations

Oksana A. Chkrebtii[*], David A. Campbell, Mark A. Girolami, Ben Calderhead

April 28, 2014

Abstract

This paper advocates expansion of the role of Bayesian statistical inference when formally quantifying uncertainty in computer models defined by systems of ordinary or partial differential equations. We adopt the perspective that implicitly defined infinite dimensional functions representing model states are objects to be inferred probabilistically. We develop a general methodology for the probabilistic integration of differential equations via model based updating of a joint prior measure on the space of functions and their temporal and spatial derivatives. This results in a posterior measure over functions reflecting how well they satisfy the system of differential equations and corresponding initial and boundary values. We show how this posterior measure can be naturally incorporated within the Kennedy and O'Hagan framework for uncertainty quantification and provides a fully Bayesian approach to model calibration. By taking this probabilistic viewpoint, the full force of Bayesian inference can be exploited when seeking to coherently quantify and propagate epistemic uncertainty in computer models of complex natural and physical systems. A broad variety of examples are provided to illustrate the potential of this framework for characterising discretization uncertainty, including initial value, delay, and boundary value differential equations, as well as partial differential equations. We also demonstrate our methodology on a large scale system, by modeling discretization uncertainty in the solution of the Navier-Stokes equations of fluid flow, reduced to over 16,000 coupled and stiff ordinary differential equations. Finally, we discuss the wide range of open research themes that follow from the work presented.

Keywords: Bayesian Numerical Analysis, Uncertainty Quantification, Gaussian Processes, Differential Equation Models, Uncertainty in Computer Models.

1 System Modelling with Differential Equations

In all of the sciences, economics and engineering there is a fundamental reliance on the use of differential equation models to describe complex phenomena concisely, using few but readily interpretable parameters, which we denote by θ. In systems of differential equations, the derivatives with respect to spatial variables, $x\in \mathcal{D} \subset \mathbb{R}^d$, and temporal variables, $t \in [a,b] \subset \mathbb{R}^+$, are related to the implicitly defined states, $\mathrm{u}(x,t) \in \mathbb{R}^P$, which are hence often analytically intractable.

The main challenge of working with differential equation models, from both mathematical and statistical perspectives, is that the solutions are generally not available in closed form. Consequently, numerical solvers are used to approximate a system solution over a discretisation grid. Numerical integration of a system of differential equations yields a deterministic approximation based on subjective choices regarding the order of the numerical method, the specified error tolerance, and the implied discretisation grid, all of which impact the quality of the approximation (Butcher, 2008; Higham, 1996). These methods implicitly make strong assumptions, for example, that small changes in the step size result in small cumulative changes in the output, and that there exists a unique, well-conditioned solution. Additionally, meaningful inferences from the model are dependent on the assumption that the approximated states contain negligible numerical error. Although the specified error tolerance and associated point-wise errors can be reduced by refining the discretisation grid, there is naturally a trade off between the computational effort involved and the accuracy of the approximated solution, which determines whether the assumption of negligible error is reasonable in

[*] oksana.chkrebtii@gmail.com; address: Department of Statistics, The Ohio State University, 1958 Neil Avenue, 404 Cockins Hall, Columbus, OH 43210-1247.

practice. Local and global numerical errors are defined point-wise and relate to the asymptotic behaviour of the deterministic approximation of the model. This form of numerical error analysis is not well suited for quantifying the functional uncertainty in the solution for the purpose of fully probabilistic model inference.

An additional challenge when working with differential equation models is that in some cases classical solution approximations provide potentially misleading trajectories. Long-time numerical solutions may be globally sensitive to truncation errors introduced at each discretisation point (e.g. Sauer et al., 1997). Ill-conditioned models can give rise to another source of uncertainty in the form of solution multiplicity (Ascher et al., 1988; Beyn and Doedel, 1981). This often cannot be readily verified analytically and clearly poses a problem for classical numerical integration methods that produce a single deterministic solution (Keller, 1968).

In this paper we address the modelling challenges that arise when using the following classes of deterministic ordinary and partial differential equation models, for which analytical solutions are generally not available. *Ordinary Differential Equation* (ODE) models implicitly represent the derivative of states $\mathrm{u}(t)$, with respect to time t through $\mathrm{u}_t(t) := \frac{d}{dt}\mathrm{u}(t) = f(t, \mathrm{u}(t), \theta)$, where we make the standard assumption that f is continuous in the first argument and Lipschitz continuous in the second argument. Inputs and boundary constraints define several model variants.

The *Initial Value Problem* (IVP) models the system states with fixed initial condition $\mathrm{u}^*(a)$, evolving according to the ODE as follows,

$$\begin{cases} \mathrm{u}_t(t) = f\big(t, \mathrm{u}(t), \theta\big), & t \in [a, b], \\ \mathrm{u}(a) = \mathrm{u}^*(a). \end{cases} \tag{1}$$

The existence of a solution is guaranteed under mild conditions (see for example, Butcher, 2008; Coddington and Levinson, 1955). Such models may be high dimensional and contain other complexities such as algebraic components, functional inputs, or higher order terms.

While IVP models specify a fixed initial condition on the system states, the *Mixed Boundary Value Problem* (MBVP) may constrain different states at different time points. Typically these constraints are imposed at the ends of the time domain giving the general form for two state mixed boundary value problems,

$$\begin{cases} \big(\mathrm{u}_t(t), \mathrm{v}_t(t)\big) = f\left(t, \big(\mathrm{u}(t), \mathrm{v}(t)\big), \theta\right), & t \in [a, b], \\ g\left(\mathrm{v}(a), \mathrm{u}(b)\right) = 0, \end{cases} \tag{2}$$

which can be straightforwardly generalised to higher dimensions and extrapolated beyond the final time point b. Whereas a unique IVP solution exists under relatively mild conditions, imposing mixed boundary constraints can result in multiple solutions (e.g. Keller, 1968) introducing severe problems for parameter estimation methods.

The *Delay Initial Function Problem* (DIFP) generalises the initial constraint of IVPs to an initial function, $\phi(t)$, thereby relating the derivative of a process to both present and past states at lags $\tau_j \in [0, \infty)$,

$$\begin{cases} \mathrm{u}_t(t) & = f\left(t, \mathrm{u}(t-\tau_1), \ldots, \mathrm{u}(t-\tau_d), \theta\right), & t \in [a, b], \\ \mathrm{u}(t) & = \phi(t), & t \in \left[a - \max_j(\tau_j), a\right]. \end{cases} \tag{3}$$

DIFPs are well suited to describing biological and physical dynamics that take time to propagate through systems. However, they pose challenges to numerical techniques due to potentially large and structured truncation error (Bellen and Zennaro, 2003). Furthermore, the sensitivity of DIFP solutions to small changes in the initial function, such as those due to interpolation error, may push an otherwise well behaved system into a chaotic regime (Taylor and Campbell, 2007).

Partial Differential Equation (PDE) models represent the derivative of states with respect to multiple arguments, for example time and spatial variables. The main classes of PDE models are based on elliptic, parabolic and hyperbolic equations. Further adding to their complexity, functional boundary constraints and initial conditions make PDE models more challenging, while their underlying theory is less developed compared to ODE models (Polyanin and Zaitsev, 2004).

1.1 Overview of the Current Landscape

Numerical discretisation error is often characterised in the form of an upper bound, referred to as *verification error* in the mathematics and engineering literature (Oberkampf and Roy, 2010). In the many cases where the error is not negligible, for example in large scale Navier-Stokes equations used in modelling fluid flow, an accurate representation of the error propagation resulting from the approximation is required. An illustrative example of quantifying this numerical error in fluid dynamics is provided by Oliver et al. (2014) and a preliminary attempt at a Bayesian formalisation of the uncertainty induced is described by Oliver and Moser (2011). The approach advocated in this paper is different in nature. We characterise discretisation uncertainty using a probabilistic representation of the solution conditional on fixed initial conditions and solver parameters. Our methodology challenges some of the assumptions used by numerical solvers and yields algorithms robust to many of the inferential issues that occur.

Increasing attention is being paid to the challenges associated with quantifying uncertainty in system models, and in particular those based on mathematical descriptions of natural or physical processes using differential equations (Ghanem and Spanos, 2003; Huttunen and Kaipio, 2007; Kaipio et al., 2004; Marzouk and Najm, 2009; Marzouk et al., 2007; Stuart, 2010). The important question of how uncertainty propagates through complex systems arises in a variety of contexts, and the focus is often on two main problems. The *forward problem* is concerned with how input uncertainty (i.e. in the parameters and initial conditions) propagates over the numerical trajectory. This type of problem has been investigated in the engineering and applied mathematics literature, making use of sampling based methods, perturbation of initial system states (Mosbach and Turner, 2009), moment closure approaches, and more recently polynomial chaos methods (see Ghanem and Spanos, 2003; Xiu, 2009, for a complete overview). The *inverse problem* concerns the uncertainty in the model inputs given measurements with error of the model outputs, and much work in this area has been presented in the statistics literature (Bock, 1983; Brunel, 2008; Calderhead and Girolami, 2011; Calderhead et al., 2009; Campbell and Lele, 2013; Campbell and Steele, 2011; Campbell and Chkrebtii, 2013; Dowd, 2007; Gugushvili and Klaassen, 2012; Ionides et al., 2006; Liang and Wu, 2008; Ramsay et al., 2007; Xue et al., 2010; Xun et al., 2014). Additionally, the inverse problem itself may be ill-conditioned when the data does not lie in the solution space of the differential equation model (Brynjarsdóttir and O'Hagan, 2014; Kennedy and O'Hagan, 2001).

The issue of discretisation uncertainty is intrinsic to both the forward and inverse problems, and it seems natural to consider the unknown solution of the differential equation itself as part of the inferential framework. There are many examples suggesting that a probabilistic functional approach is well suited for modelling solution uncertainty for differential equation models; indeed solving PDEs involves estimation of a field, DIFPs require estimation of an unknown function, and MBVPs may have multiple solutions. All of these problems may be naturally framed in terms of measures over a function space (Stuart, 2010). A Bayesian functional approach offers a way of defining and updating measures over possible trajectories of a system of differential equations and propagating the resulting uncertainty consistently through the inferential process. The mathematical framework for defining numerical problems on function spaces has been developed through the foundational work of Stuart (2010), Kaipio and Somersalo (2007), and Skilling (1991), and this is the natural setting that we adopt in our contribution. We develop these ideas further and investigate general approaches to working with classes of differential equations from a Bayesian perspective.

The use of probability in numerical analysis has a long history. In the 1950s, Monte Carlo methods were originally developed to estimate analytically intractable high dimensional integrals, and extensions of this methodology have since had an immense impact in a wide variety of areas. Markov chain Monte Carlo (MCMC) methods have allowed the rapid development and application of Bayesian statistical approaches to quantifying uncertainty in scientific and engineering problems. Theoretical analyses of Bayesian statistical approaches to classical numerical analysis problems appear to date back all the way to Poincaré, as eloquently summarised by Diaconis (1988). More recently, O'Hagan (1992); Skilling (1991) also provided practical motivations for developing and employing Bayesian techniques to characterise the uncertainty that results from computational restrictions in the case of integration, since we cannot evaluate a function at every point in its input space and must therefore account for discretisation uncertainty. Such work is closely related to the idea of computer emulation of large and complex computer models; the use of Gaussian processes to model functions that are computationally very expensive to evaluate has found application in many challenging areas of science, such as climate prediction and geophysics (Conti and O'Hagan, 2010; Kennedy and O'Hagan, 2001; Tokmakian et al., 2012). Other recent examples of applying Bayesian approaches to numerical integration include Hennig and Hauberg (2014), who apply Skilling's approach to the calculation of Riemannian statistics, and Kennedy (1998); Osborne et al. (2012), whose methods allow for a more informed probabilistic approach to the choice of integration points, based on the predicted variance of the surrogate model given limited function evaluations. However, such approaches assume no error on the function evaluation itself, and are therefore not applicable to differential equation models, where the function evaluation gives only an approximate derivative of the solution of the system at the next discrete time point, based on the current predictive distributions over the states.

1.2 Contributions of the Paper

The aim of this paper is to make a case for probabilistically characterising discretisation error when solving general classes of differential equations, and for viewing this epistemic uncertainty as an important and integral part of the overall model inference framework.

We first develop a probabilistic differential equation solver that characterises discretisation uncertainty in the solution, by formalising and substantially extending the ideas first presented by Skilling (1991). The sequential sampling approach suggested is made computationally feasible through recursive Bayesian updating, and it closely follows the sequential construction of a proof of consistency that we present in the Appendix, ensuring the algorithm converges to the exact solution as the time-step tends to zero. In our approach, we treat the sequential model evaluations as auxiliary parameters, over which the solution can then be marginalised, and we further extend the basic method to the cases of mixed boundary value problems exhibiting multiplicity of solutions, delay initial function problems with partially observed initial functions, stiff and chaotic PDEs via spectral projection, as well as providing a framework for directly solving PDEs. We give an example of the scalability of our approach using a Navier-Stokes system involving the solution of over 16,000 coupled stiff ordinary differential equations. Finally, we adopt a flexible and general forward-simulation approach that allows us to embed the formal quantification of discretisation uncertainty for differential equations within the Kennedy and O'Hagan framework, hence defining it as part of the full inferential procedure for inverse problems. We provide code for our Probabilistic Differential Equation Solver (PODES), which allows replication of all results presented in this paper. This is available at http://web.warwick.ac.uk/PODES.

1.3 Structure of the Paper

We begin in Section 2 by discussing model discrepancy within the Bayesian approach to uncertainty quantification, as first presented by Kennedy and O'Hagan (2001) and subsequently adopted in practice throughout the statistics literature. We discuss how existing numerical approaches to differential equation integration ignore discretisation uncertainty when the solution is not available in closed form. We then develop a general Bayesian framework for modelling this epistemic uncertainty in differential equations, inspired by the work of Skilling (1991) and O'Hagan (1992). We begin in Section 3 by defining functional priors on the states and derivatives for general ordinary and partial differential equations. Then, in Section 4, we develop a Bayesian approach for updating our prior beliefs given model information obtained sequentially and self-consistently over a given discretisation grid, thus probabilistically characterising the finite dimensional representation of an underlying infinite dimensional solution given model parameters and initial conditions. We show that the resulting posterior trajectory converges in probability to the exact solution under standard assumptions. We then incorporate this methodology into the inverse problem in Section 5, and provide examples in Section 6. Finally, we discuss our proposed approach to Bayesian uncertainty quantification for differential equation models and highlight open areas of research. The Appendix contains proofs and further mathematical and algorithmic details.

2 Quantifying Uncertainty in Differential Equation Models

The importance of formally quantifying sources of uncertainty in computer simulations when mathematically modelling complex natural phenomena, such as the weather, ocean currents, ice sheet flow, and cellular protein transport, is widely acknowledged. These model based simulations inform the reasoning process when, for example, assessing financial risk in deciding on oil field bore configurations, or forming government policy in response to extreme weather events. As such, accounting for all sources of uncertainty and propagating them in a coherent manner throughout the entire inference and decision making process is of great importance.

Consider data, $y(\mathbf{t})$, observed at discrete time points $\mathbf{t} = [t_1, t_2, \dots, t_T]$ and a set of model parameters θ. Using the exact solution of a mathematical model represented by $\mathrm{u}^*(\mathbf{t}, \theta)$, a simplified observation model based on some measurement error structure $\epsilon(\mathbf{t})$ is,

$$y(\mathbf{t}) = \mathrm{u}^*(\mathbf{t}, \theta) + \epsilon(\mathbf{t}). \tag{4}$$

In full generality, a nonlinear transformation $\mathcal{G}$ of $\mathrm{u}^*(\mathbf{t}, \theta)$ may be observed, however for expositional clarity we assume this is an identity. In our setting, $\mathrm{u}^*(\mathbf{t}, \theta)$ is the unique function satisfying a general system of differential equations. When performing statistical inference over such models, we are interested in the joint posterior distribution over all unknowns, possibly including the initial and boundary conditions. Throughout the paper, θ will be augmented to contain all the variables of interest.

In the landmark paper of Kennedy and O'Hagan (2001) a number of sources of uncertainty were identified when modelling a natural or physical process. Their acknowledgement of uncertainty from incomplete knowledge of the process and the corresponding model inadequacy motivates a probabilistic view of a model-reality mismatch $\delta(\mathbf{t})$ also studied in recent papers by Brynjarsdóttir and O'Hagan (2014); Huttunen and Kaipio (2007); Kaipio and Somersalo (2007); and Stuart (2010). In the engineering literature, this form of error is also know as *validation error* (see for example, Oberkampf and Roy, 2010). Following Kennedy and O'Hagan by defining $\delta(\mathbf{t})$ as a random function drawn from a Gaussian Process (GP), the observational model becomes,

$$y(\mathbf{t}) = \mathrm{u}^*(\mathbf{t}, \theta) + \delta(\mathbf{t}) + \epsilon(\mathbf{t}). \tag{5}$$

Due to the lack of an analytical solution for most nonlinear differential equations, the likelihood $p\big(y(\mathbf{t}) \mid \mathrm{u}^*(\mathbf{t}, \theta), \theta\big)$ cannot be obtained in closed form. This issue is dealt with throughout the statistics literature by replacing the exact likelihood with a surrogate, $p\big(y(\mathbf{t}) \mid \hat{\mathrm{u}}^N(\mathbf{t}, \theta), \theta\big)$, based on an N-dimensional approximate solution, $\hat{\mathrm{u}}^N(\mathbf{t}, \theta)$, obtained using numerical integration methods (for example, a Runge-Kutta solver with N time steps) whose accuracy has been well studied (see for example, Henrici, 1964). However, there are still many scenarios for which standard approaches for characterising upper bounds on numerical error, and subsequently propagating this uncertainty throughout the rest of the inferential process are unsatisfactory. Limited computation and coarse mesh size are contributors to the numerical error, as described in the comprehensive overview of Oberkampf and Roy (2010). The seemingly innocuous assumption of negligible numerical integration error can subsequently lead to serious statistical bias and misleading inferences for certain classes of differential equation models, and we illustrate this point in Section 6.1 using a simple example. We may represent this additional uncertainty using the term $\zeta(\mathbf{t}, \theta) = \mathrm{u}^*(\mathbf{t}, \theta) - \hat{\mathrm{u}}^N(\mathbf{t}, \theta)$, such that,

$$y(\mathbf{t}) = \hat{\mathrm{u}}^N(\mathbf{t}, \theta) + \zeta(\mathbf{t}, \theta) + \delta(\mathbf{t}) + \epsilon(\mathbf{t}). \tag{6}$$

Kennedy and O'Hagan (2001) suggest modelling the computer code using a Gaussian process that is agnostic to the specific form of the underlying mathematical model, which is therefore considered as a "black box". In the discussion of their paper, H. Wynn argues that the sensitivity equations and other underlying mathematical structures that govern the computer simulation could also be included in the overall uncertainty analysis. In the current paper, we "open the black box" by explicitly modelling the solution and associated discretisation uncertainty, $\hat{\mathrm{u}}^N(\mathbf{t}, \theta) + \zeta(\mathbf{t}, \theta)$. This allows the Kennedy and O'Hagan framework to be further enriched by incorporating detailed knowledge of the mathematical model being employed. Noting that the posterior measure over the computer model is defined independently of the observed data suggests a further construction by probabilistically characterising the uncertainty due to system states being defined implicitly by differential equations. This leads to the development of a fully probabilistic scheme for solving differential equations and suggests a powerful new approach to modelling uncertainty in computer models.

We model uncertainty in a finite dimensional representation of the infinite dimensional solution through a probability statement on a space of suitably smooth functions. Restricting ourselves to Hilbert spaces for modelling our knowledge of $\mathrm{u}^*(\mathbf{t},\theta)$, we define a Gaussian prior measure on the function space (Stuart, 2010). We then directly model our knowledge about the solution via the stochastic process $\mathrm{u}(\mathbf{t},\theta)$, thus replacing (6) with $y(\mathbf{t}) = \mathrm{u}(\mathbf{t},\theta) + \delta(\mathbf{t}) + \epsilon(\mathbf{t})$.

The contribution in this paper is in our definition and use of $\mathrm{u}(\mathbf{t},\theta)$, therefore for expositional clarity we focus our attention on the joint posterior measure over differential equation model states, parameters, and associated auxiliary parameters, Ψ, of our probabilistic model of uncertainty,

$$p\big(\theta, \mathrm{u}(\mathbf{t},\theta), \Psi \mid y(\mathbf{t}), N\big) \propto \underbrace{p\big(y(\mathbf{t}) \mid \mathrm{u}(\mathbf{t},\theta), \theta\big)}_{\text{Likelihood}} \times \underbrace{p\big(\mathrm{u}(\mathbf{t},\theta) \mid \theta, \Psi, N\big)}_{\text{Model}} \times \underbrace{p\big(\theta, \Psi\big)}_{\text{Prior}}. \tag{7}$$

Our Bayesian probabilistic integration framework for differential equation models fully explores the space of trajectories that approximately satisfy model dynamics under theoretical guarantees of consistency. The framework presented in this paper is demonstrated on models described using nonlinear ordinary, delay, mixed boundary value and partial differential equations, and we explicitly address multiplicity of solutions in Section 6.2. For notational simplicity we hereafter omit the dependence of u on θ.

3 Gaussian Prior Measure for States and Derivatives Defined by a Differential Equation Model

When considering uncertainty in the infinite dimensional solution of differential equations, the natural measure for spaces of a wide class of functions is Gaussian (Stuart, 2010). As such we define a Gaussian process (GP) prior measure jointly on the state u and its time derivative u_t. We will now discuss the prior construction for ordinary differential equations, and then consider the case for partial differential equations.

In the remainder of the paper, we will denote by R_λ a deterministic, square integrable kernel function with length-scale $\lambda \in (0,\infty)$, and its integrated version by $\mathrm{Q}_\lambda(t_1,t_2) = \int_a^{t_1} \mathrm{R}_\lambda(s,t_2)\mathrm{d}s$. Here a represents the lower boundary of our temporal domain. Our model for the derivative has covariance operator $\mathrm{cov}(\mathrm{u}_t(t_1), \mathrm{u}_t(t_2)) = \alpha^{-1}\int_{\mathbb{R}} \mathrm{R}_\lambda(t_1,s)\mathrm{R}_\lambda(t_2,s)\mathrm{d}s := \mathrm{RR}(t_1,t_2)$, where α is a prior precision parameter. Therefore the covariance on the state is its integrated version, $\mathrm{cov}(\mathrm{u}(t_1), \mathrm{u}(t_2)) = \alpha^{-1}\int_{\mathbb{R}} \mathrm{Q}_\lambda(t_1,s)\mathrm{Q}_\lambda(t_2,s)\mathrm{d}s := \mathrm{QQ}(t_1,t_2)$. The cross covariance terms are defined in a similar manner and denoted as $\mathrm{RQ}(t_1,t_2)$ and $\mathrm{QR}(t_1,t_2)$ respectively. We note that $\mathrm{RQ}(t_1,t_2) = \mathrm{QR}^\dagger(t_1,t_2)$, where $\dagger$ represents the adjoint.

We assume a joint Gaussian prior measure on the state and its derivative,

$$\begin{bmatrix}\mathrm{u}_t \\ \mathrm{u}\end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix}\mathrm{m}_t \\ \mathrm{m}\end{bmatrix}, \begin{bmatrix}\mathrm{RR}(t_1,t_2) & \mathrm{QR}^\dagger(t_1,t_2) \\ \mathrm{QR}(t_1,t_2) & \mathrm{QQ}(t_1,t_2)\end{bmatrix}\right). \tag{8}$$

The Gaussian conditional prior measures for e.g. $p\left(\mathrm{u}_t(t) \mid \mathrm{u}(t)\right)$ take the standard forms (Stuart, 2010). The choice of prior means and covariance structure, as well as the impact and choice of auxiliary parameters, $\Psi = [\alpha, \lambda]$, are discussed in the Appendix (Section 8.2). For exposition, we take the auxiliary parameters Ψ as fixed, although we note that propagating a distribution over Ψ through the probabilistic solver is straightforward, and estimation of Ψ is addressed in Section 6.4.

This model straightforwardly generalises to ODE problems of order greater than one, by defining the prior jointly on the state, u, and any required derivatives. Alternatively, higher order ODE problems may be restated in the first order by introducing additional states. We have found that adding states to the model is computationally straightforward, so we adopt this approach in practice, working with the prior in equation (8).

PDE problems require a prior measure over multivariate trajectories, as well as modelling single or higher order derivatives with respect to spatial inputs. Therefore the prior specification will depend on the PDE model. Consider as an illustrative example the parabolic heat equation, modelling the heat diffusion over time along a single spatial dimension by,

$$\left\{\begin{array}{lcll} \mathrm{u}_t(x,t) & = & \kappa\mathrm{u}_{xx}(x,t), & t \in [0, 0.25],\ x \in [0,1] \\ \mathrm{u}(x,t) & = & \sin(x\pi), & t = 0,\ x \in [0,1], \\ \mathrm{u}(x,t) & = & 0, & t \in [0, 0.25],\ x = 0, 1. \end{array}\right. \tag{9}$$

One modelling choice for incorporating the spatial component of the PDE into the covariance is to adopt a product structure. The covariance over time is defined as above, and over space may be defined using a similar construction. Let R_μ be a kernel function with length-scale μ. Define $\mathrm{Q}_\mu(x_1, x_2) = \int_c^{x_1} \mathrm{R}_\mu(z, x_2)\mathrm{d}z$ and $\mathrm{S}_\mu(x_1, x_2) = \int_c^{x_1} \mathrm{Q}_\mu(z, x_2)\mathrm{d}z$, where c denotes a spatial lower boundary (in (9), this is $c = 0$). The spatial covariance structures are defined similarly to the temporal form in the ODE example as,

$$\begin{aligned}
\mathrm{cov}(\mathrm{u}_{xx}(x_1, t_1), \mathrm{u}_{xx}(x_2, t_2)) &= \beta^{-1} \textstyle\int_{\mathbb{R}} \mathrm{R}_\mu(x_1, z)\mathrm{R}_\mu(x_2, z)\mathrm{d}z \; \mathrm{QQ}(t_1, t_2) := \mathrm{RR}(x_1, x_2)\mathrm{QQ}(t_1, t_2), \\
\mathrm{cov}(\mathrm{u}(x_1, t_1), \mathrm{u}(x_2, t_2)) &= \beta^{-1} \textstyle\int_{\mathbb{R}} \mathrm{S}_\mu(x_1, z)\mathrm{S}_\mu(x_2, z)\mathrm{d}z \; \mathrm{QQ}(t_1, t_2) := \mathrm{SS}(x_1, x_2)\mathrm{QQ}(t_1, t_2),
\end{aligned}$$

where β is a spatial prior precision parameter. Cross covariances are defined analogously. The prior construction follows by defining a product structure for space and time,

$$\begin{bmatrix} \mathrm{u}_{xx} \\ \mathrm{u}_t \\ \mathrm{u} \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} \mathrm{m}_{xx} \\ \mathrm{m}_t \\ \mathrm{m} \end{bmatrix}, \begin{bmatrix} \mathrm{RR}(x_1, x_2)\mathrm{QQ}(t_1, t_2) & \mathrm{SR}^\dagger(x_1, x_2)\mathrm{QR}(t_1, t_2) & \mathrm{SR}^\dagger(x_1, x_2)\mathrm{QQ}(t_1, t_2) \\ \mathrm{SR}(x_1, x_2)\mathrm{QR}^\dagger(t_1, t_2) & \mathrm{SS}(x_1, x_2)\mathrm{RR}(t_1, t_2) & \mathrm{SS}(x_1, x_2)\mathrm{QR}^\dagger(t_1, t_2) \\ \mathrm{SR}(x_1, x_2)\mathrm{QQ}(t_1, t_2) & \mathrm{SS}(x_1, x_2)\mathrm{QR}(t_1, t_2) & \mathrm{SS}(x_1, x_2)\mathrm{QQ}(t_1, t_2) \end{bmatrix} \right). \tag{10}$$

In the next section, we describe in detail the framework for sequentially linking the prior in equation (8) with the ODE model. We describe the corresponding algorithm for updating the prior in equation (10) under the heat equation PDE in the Appendix (Section 8.1). We provide examples illustrating its use experimentally in Section 6.

3.1 Prior Specification

The joint prior on the state and its derivative is defined in equation (8) in terms of mean functions m, m_t and a covariance structure determined by the choice of kernel function R_λ, which is parameterised by the auxiliary variables α and λ. The choice of the covariance kernel should reflect our assumptions regarding the smoothness of the exact but unknown differential equation solution. In the Appendix (Section 8.3), we provide covariance structures based on two kernels: the infinitely differentiable squared exponential and the non-differentiable uniform kernel. Gaussian process models typically match the kernel for a given application to prior information about smoothness of the underlying function (see, for example, Rasmussen and Williams, 2006). It is also important to be aware that imposing unrealistically strict smoothness assumptions on the state space by choice of covariance structure may introduce estimation bias if the exact solution is not at least as smooth. Therefore, in cases where the solution smoothness is not known a priori, one can err on the side of caution by using less regular kernels. In the examples reported in this paper we chose to work with stationary derivative covariance structures for simplicity, however we point out that there will be classes of problems where non-stationary kernels may be more appropriate and can likewise be incorporated into the Gaussian process framework. The mean function is chosen based on any prior knowledge about how the system evolves over time. Typically, however, such prior detailed knowledge is unavailable, in which case it is reasonable to use a constant mean function. Additionally, we condition on the known boundary values, enforcing them by choice of the prior mean. For example, for IVPs, we satisfy the boundary constraint exactly by choosing a differentiable prior mean function m such that $\mathrm{m}(a) := \mathrm{u}^*(a)$.

Now that the form of the prior measure has been defined, the corresponding posterior is obtained in the following two sections in a sequential manner. This iterative updating closely follows the sequential structure of the proof of consistency, provided in the Appendix (Section 8.6). Here we point out that, although the amount and quality of prior information regarding the true solution will affect the efficiency of our probabilistic solvers, we can still expect convergence as the time step tends to zero, subject to some standard assumptions on the kernel function. Furthermore, in Section 6.1, we illustrate the extent of prior influence on the probabilistic solution for the heat equation PDE. We find that the use of a zero-mean prior on the spatial and temporal derivatives has minimal impact on the solution, even for reasonably rough discretisation grids.

4 Probabilistic Integration for Systems of Differential Equations

We now introduce a sequential framework to characterise the epistemic model uncertainty component in equation (7). We restrict our attention to the ODE initial value problem (1) and consider more general ODE and PDE problems in Section 6. We model discretisation uncertainty by updating the joint prior on the unknown state and derivative iteratively, by evaluating the deterministic ODE model at a finite number N of grid points over the temporal domain, which we denote by $\mathbf{f}_{1:N}$. Algorithm 1 produces a sample from the joint distribution, $[\mathrm{u}, \mathbf{f}_{1:N} \mid \theta, \Psi, N]$.

Let us consider the problem of modelling uncertainty in the exact but unknown solution of a nonlinear ODE initial value problem. We are given the differential equation model, $\mathrm{u}_t(t) = f(t, \mathrm{u}(t), \theta)$, which implicitly defines the derivatives in terms of a Lipschitz continuous function f of the states given θ and a fixed initial value, $\mathrm{u}^*(a)$. We model our uncertainty regarding the unique explicit state, u, and its derivative, u_t, satisfying the initial value problem, via the joint GP prior measure in equation (8). The following procedure sequentially links the prior on the state with the ODE model defined by f.

Consider a discretisation grid made up of N time points $\mathbf{s} := [s_1, \ldots, s_N]$ on the interval $[a, b]$ such that $a = s_1 \leq \cdots \leq s_N = b$. We begin by fixing the known initial value, $\mathrm{u}(s_1) := \mathrm{u}^*(a)$, and computing the exact derivative $\mathrm{f}_1 := f(s_1, \mathrm{u}(s_1), \theta)$ at s_1, via the deterministic ODE model. We then update our joint GP prior given the computed exact derivative f_1, obtaining the conditional predictive distribution for the state at the subsequent grid location s_2,

$$p(\mathrm{u}(s_2) \mid \mathrm{f}_1, \Psi) = \mathcal{N}\big(\mathrm{u}(s_2) \mid \mathrm{m}(s_2), \mathrm{C}(s_2, s_2)\big), \tag{11}$$

with,

$$\begin{aligned} \mathrm{m}(s_2) &= \mathrm{QR}(s_2, s_1)\mathrm{RR}(s_1, s_1)^{-1}\mathrm{f}_1, \\ \mathrm{C}(s_2, s_2) &= \mathrm{QQ}(s_2, s_2) - \mathrm{QR}(s_2, s_1)\mathrm{RR}(s_1, s_1)^{-1}\mathrm{QR}(s_2, s_1). \end{aligned}$$

This predictive distribution describes our current uncertainty about the solution at time s_2. We now sample a realisation, $\mathrm{u}(s_2)$, of the predictive process, and again link our prior to the deterministic ODE model by computing $\mathrm{f}_2 := f(s_2, \mathrm{u}(s_2), \theta)$. In contrast to the first time point s_1, we can no longer guarantee that the realisation at the second time point, $\mathrm{u}(s_2)$, and its derivative, $\mathrm{u}_t(s_2)$, exactly satisfy the ODE model, i.e. that $\mathrm{u}_t(s_2) = \mathrm{f}_2$. Therefore, at time s_2 we explicitly model the mismatch between the ODE evaluation f_2 and the process derivative, $\mathrm{u}_t(s_2)$, as,

$$p\big(\mathrm{u}_t(s_2) \mid \mathrm{f}_2, \Psi\big) = \mathcal{N}\big(\mathrm{u}_t(s_2) \mid \mathrm{f}_2, \mathrm{C}_t(s_2, s_2)\big),$$

where the magnitude of the mismatch may be described by the variance, $\mathrm{C}_t(s_2, s_2)$, of the predictive posterior over the derivative given by,

$$\mathrm{C}_t(s_2, s_2) = \mathrm{RR}(s_2, s_2) - \mathrm{RR}(s_2, s_1)\mathrm{RR}(s_1, s_1)^{-1}\mathrm{RR}(s_1, s_2).$$

For systems in which we believe this mismatch to be strongly non-Gaussian, we may appropriately modify the above model and associated sampling strategy, as described in the Discussion section. We may now update our joint posterior from the previous iteration by conditioning on the augmented vector $\mathbf{f}_{1:2} := [\mathrm{f}_1, f(s_2, \mathrm{u}(s_2), \theta)]$. We therefore define the matrix $\Lambda_{2\times 2} := \mathrm{diag}\{0, \mathrm{C}_t(s_2, s_2)\}$ to describe the mismatch between the process derivative and the ODE function evaluations at s_1 and s_2. The new predictive posterior,

$$p(\mathrm{u}(s_3) \mid \mathbf{f}_{1:2}, \Psi) = \mathcal{N}\big(\mathrm{u}(s_3) \mid \mathrm{m}(s_3), \mathrm{C}(s_3, s_3)\big),$$

has mean and covariance,

$$\mathrm{m}(s_3) = \mathrm{QR}(s_3, \mathbf{s}_{1:2})\big(\mathrm{RR}(\mathbf{s}_{1:2}, \mathbf{s}_{1:2}) + \Lambda_{2\times 2}\big)^{-1}\mathbf{f}_{1:2},$$

$$\mathrm{C}(s_3, s_3) = \mathrm{QQ}(s_3, s_3) - \mathrm{QR}(s_3, \mathbf{s}_{1:2})\big(\mathrm{RR}(\mathbf{s}_{1:2}, \mathbf{s}_{1:2}) + \Lambda_{2\times 2}\big)^{-1}\mathrm{QR}(s_3, \mathbf{s}_{1:2})^\top.$$

As in the previous two steps, we sample the state realisation $\mathrm{u}(s_3)$ from the above predictive posterior distribution. We then apply the deterministic transformation f, to obtain $\mathrm{f}_3 := f(s_3, \mathrm{u}(s_3), \theta)$, whose mismatch with the realised derivative $\mathrm{u}_t(s_3)$ is again modelled as,

$$p\big(\mathrm{u}_t(s_3) \mid \mathrm{f}_3, \Psi\big) = \mathcal{N}\big(\mathrm{u}_t(s_3) \mid \mathrm{f}_3, \mathrm{C}_t(s_3, s_3)\big).$$

We next augment $\mathbf{f}_{1:3} := [\mathbf{f}_{1:2}, \mathrm{f}_3]$. The corresponding mismatch matrix therefore has a diagonal structure, $\Lambda_{3\times 3} := \mathrm{diag}\{\Lambda_{2\times 2}, \mathrm{C}_t(s_3, s_3)\}$, where the step-ahead predicted covariance in the derivative space is,

$$\mathrm{C}_t(s_3, s_3) = \mathrm{RR}(s_3, s_3) - \mathrm{RR}(s_3, \mathbf{s}_{1:2})\big(\mathrm{RR}(\mathbf{s}_{1:2}, \mathbf{s}_{1:2}) + \Lambda_{2\times 2}\big)^{-1}\mathrm{RR}(\mathbf{s}_{1:2}, s_3)^\top.$$

The diagonal elements of Λ, which are step-ahead predictive derivative variances, are non-decreasing, reflecting our growing uncertainty about how well the realised state obeys the ODE model as we take our sample further and further away from the known initial state. It is also shown in the Appendix (Section 8.7) that its elements tend to zero with the step size but at a much faster rate than the step size. The general scheme can be written according to Algorithm 1, and an illustration of this is provided in Figure 1.

Algorithm 1 Sample from the joint posterior distribution of u and $\mathbf{f}_{1:N}$ for an ODE initial value problem given θ, Ψ, N

```
At time $s_1 := a$, initialise the derivative $\mathrm{f}_1 := f\big(s_1, \mathrm{u}(s_1), \theta\big)$ for initial state $\mathrm{u}(s_1) := \mathrm{u}^*(a)$, and define associated model-derivative mismatch, $\Lambda_{1\times 1} := 0$;
for $n = 1 : N - 1$ do
    Define the predictive state mean and variance,
        $\mathrm{m}(s_{n+1}) = \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathbf{f}_{1:n},$
        $\mathrm{C}(s_{n+1}, s_{n+1}) = \mathrm{QQ}(s_{n+1}, s_{n+1}) - \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})^\top;$
    Sample step-ahead realisation $\mathrm{u}(s_{n+1})$ from the predictive distribution of the state,
        $p\big(\mathrm{u}(s_{n+1}) \mid \mathbf{f}_{1:n}, \Psi\big) = \mathcal{N}\big(\mathrm{u}(s_{n+1}) \mid \mathrm{m}(s_{n+1}), \mathrm{C}(s_{n+1}, s_{n+1})\big);$
    Evaluate the ODE model $\mathrm{f}_{n+1} := f(s_{n+1}, \mathrm{u}(s_{n+1}), \theta)$ for realisation $\mathrm{u}(s_{n+1})$ at the subsequent grid point, $s_{n+1}$, and augment the vector $\mathbf{f}_{1:n+1} := [\mathbf{f}_{1:n}, \mathrm{f}_{n+1}]$;
    Define the predictive derivative variance,
        $\mathrm{C}_t(s_{n+1}, s_{n+1}) = \mathrm{RR}(s_{n+1}, s_{n+1}) - \mathrm{RR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{RR}(\mathbf{s}_{1:n}, s_{n+1})^\top,$
    and augment the matrix $\Lambda_{(n+1)\times(n+1)} := \mathrm{diag}\big\{\Lambda_{n\times n}, \mathrm{C}_t(s_{n+1}, s_{n+1})\big\}$;
end for

Define,
    $\mathrm{m}(\cdot) = \mathrm{QR}(\cdot, \mathbf{s}_{1:N})\big(\mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda_{N\times N}\big)^{-1}\mathbf{f}_{1:N},$
    $\mathrm{C}(\cdot, \cdot) = \mathrm{QQ}(\cdot, \cdot) - \mathrm{QR}(\cdot, \mathbf{s}_{1:N})\big(\mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda_{N\times N}\big)^{-1}\mathrm{QR}(\cdot, \mathbf{s}_{1:N})^\top;$
Return both $\mathrm{u} \sim \mathcal{GP}(\mathrm{m}, \mathrm{C})$ and $\mathbf{f}_{1:N}$.
```

We emphasise that integration of the ODE model proceeds probabilistically via Gaussian process integration, without the use of numerical approximations. We obtain a posterior distribution over trajectories

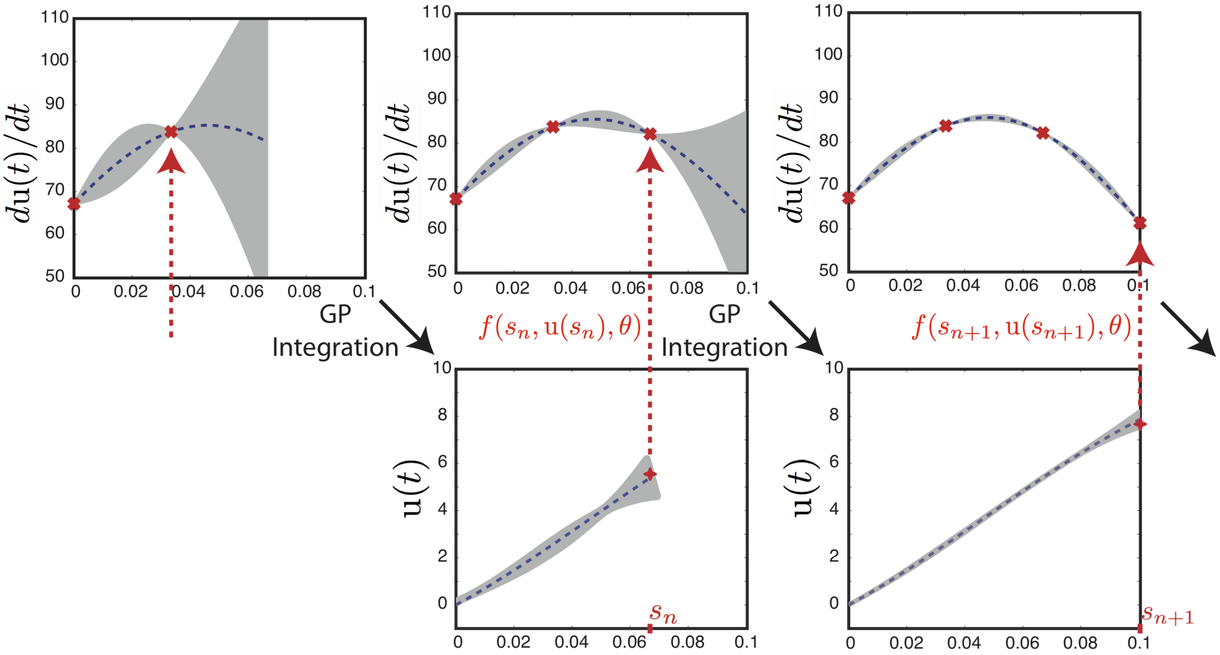

Figure 1: Illustration of Algorithm 1 for generating a sample from the joint distribution of derivative observations and possible trajectories with density $p\big(\mathrm{u}(\mathbf{t}), \mathbf{f}_{1:N} \mid \theta, \Psi\big)$. Given two derivative model realisations (red points), we obtain a posterior distribution over the derivative space (top left) and over the state space (bottom middle). A sample is then drawn from the predictive posterior over the states at the next time point s_n (bottom middle), and a model realisation is obtained by mapping $\mathrm{u}(s_n)$ to the derivative space via the function f (top middle, rightmost red point). Given these three model evaluations, this procedure may be repeated (bottom right, top right) in an analogous manner.

u that are governed by the differential equation model over a discrete number of grid points. In the Appendix (Section 8.6), we provide a proof that under certain conditions, the posterior process $\mathrm{u}(t)$ obtained via Algorithm 1 tends to the unique solution $\mathrm{u}^*(t)$ satisfying IVP (1), as follows,

$$\mathrm{E}\big\{|\mathrm{u}(t) - \mathrm{u}^*(t)| \;\big|\; \theta, \Psi\big\} \to 0, \quad \text{as } \alpha^{-1}, \lambda, h \to 0,$$

where h is the maximum length between consecutive discretisation grid points.

5 Fully Bayesian Posterior Inference for the Inverse Problem

We now present a framework to propagate the model discretisation uncertainty characterised in Section 4 through the inverse problem, where we wish to infer model parameters from measurement data. We describe one possible Markov chain Monte Carlo procedure that generates a sample from the joint posterior distribution of the state at the data locations $\mathbf{t} = [t_1, \cdots, t_T]$ and the unknown parameters θ conditional on noisy observations $y(\mathbf{t}) = \mathcal{G}\big(\mathrm{u}(\mathbf{t}), \theta\big) + \epsilon(\mathbf{t})$ of the states. We assume the parameters θ also include any unknown initial conditions, or auxiliary parameters. Once again, for expositional simplicity, we focus on direct observations of states governed by an ODE initial value problem. However, extension to other ODE and PDE problems is straightforward given a probabilistic solution. We provide an application where states are indirectly observed through the nonlinear transformation of states $\mathcal{G}$ in Section 6.4.

Algorithm 2 targets the posterior density in equation (7) by forward model proposals via conditional simulation from Algorithm 1. This proposal step avoids the need to explicitly calculate the intractable marginal density $\int p\big(\mathrm{u}(\mathbf{t}), \mathbf{f}_{1:N} \;\big|\; \theta, \Psi, N\big)\, \mathrm{d}\mathbf{f}_{1:N}$, and can be implemented efficiently as described in the Appendix (Section 8.1). Such partially likelihood-free MCMC implementations (Marjoram et al., 2003) are widely used in the area of inference for stochastic differential equations (see, for example, Golightly and Wilkinson, 2011) for simulating sample paths within the inverse problem.

Algorithm 2 Draw K samples from the posterior distribution with density $p\big(\theta, \mathrm{u}(\mathbf{t}) \mid y(\mathbf{t}), \Psi\big)$

```
Initialise $\theta$ and conditionally sample a realisation of the state $\mathrm{u}(\mathbf{t})$ via Algorithm 1;
for $k = 1 : K$ do
    Propose $\theta' \sim q(\theta' \mid \theta)$, where $q$ is a proposal density;
    Sample a probabilistic realisation of the state $\mathrm{u}'(\mathbf{t})$ conditioned on $\theta'$ via Algorithm 1;
    Compute:
        $\rho = \frac{q(\theta' \mid \theta)}{q(\theta \mid \theta')} \frac{p(\theta')}{p(\theta)} \frac{p\big(y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}'(\mathbf{t}), \theta'\big), \Sigma\big)}{p\big(y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}(\mathbf{t}), \theta\big), \Sigma\big)}$;
    if $\min\{1, \rho\} > \mathrm{U}[0, 1]$ then
        Update $\theta = \theta'$;
        Update $\mathrm{u}(\mathbf{t}) = \mathrm{u}'(\mathbf{t})$;
    end if
    Return $\theta, \mathrm{u}(\mathbf{t})$.
end for
```

We now have all the components to take a fully Bayesian approach for quantifying uncertainty on differential equation models of natural and physical systems.

6 Forward Simulation and Inference Examples

In this section we use the framework developed in the previous two sections in applications to a wide range of systems, including ODE and PDE boundary value problems and delay initial function problems.

6.1 Posterior Inference of Conductivity Parameter in a Parabolic PDE

As an illustrative example, we demonstrate the use of our probabilistic framework on the heat equation presented in (9). We model our uncertainty about the solution through the prior (10) defined over time and space using a product covariance structure as described in Section 3. We use our probabilistic framework to integrate the state over time at each of the spatial discretization grid points, employing a uniform covariance kernel for the time component and a squared exponential covariance kernel for the spatial component. We therefore characterise the discretisation uncertainty in both the time and spatial domains by conditioning on PDE model evaluations corresponding to approximate time derivatives at each discretisation grid location, dependent on sequentially sampled realisations from the predictive posterior distribution over the second order spatial derivatives. We describe the full algorithmic construction in the Appendix (Algorithm 3).

In the following numerical simulations, we consider dynamics with $\kappa = 1$ and initial function $\mathrm{u}^*(x, 0) = \sin(x\pi)$, $x \in [0, 1]$. We firstly consider the probabilistic forward problem using a variety of discretisation grid sizes. In Figure 2 we compare the exact solution with the probabilistic solution on two different grids; a coarse discretisation of 15 points in the spatial domain and 50 points in the temporal domain, and a finer discretisation of 29 points in the spatial domain and 100 points in the temporal domain. We observe from the simulations that as the mesh size becomes finer, the uncertainty in the solution decreases, in agreement with the consistency result (Appendix, Section 8.6), where it is shown that the probabilistic solution should tend to the exact solution as the grid spacing tends to zero.

Although this is an illustrative example using a simple toy system, the characterisation of spatial uncertainty is vital for more complex models, where there are computational constraints limiting the number of system evaluations that may be performed. We can see the effect of such discretisation uncertainty by performing posterior inference for the parameter κ given data simulated from an exact solution with $\kappa = 1$. Figure 3 shows the posterior distribution over κ obtained by using both a "forward in time, centred in space" (FTCS) finite difference solver and a probabilistic solver under a variety of discretisation grids. The use of a deterministic solver illustrates the problem of inferential bias and overconfident posterior variance that may occur if discretisation uncertainty is not taken into account and too coarse a grid is employed. In this illustrative setting, the use of a probabilistic solver propagates discretisation uncertainty in the solution through to the posterior distribution over the parameters. We obtain parameter estimates that assign positive probability mass to the true value of κ, even when using a coarsely discretised grid, avoiding the problem of overconfident parameter inferences that exclude the true value.

6.2 Solution multiplicity in a mixed boundary value problem

MBVPs introduce challenges for many existing numerical solvers, which typically rely on optimisation and the theory of IVPs to estimate the unspecified initial state, $\mathrm{u}(a)$ in (2). The optimisation over $\mathrm{u}(a)$ is performed until the corresponding IVP solution satisfies the specified boundary condition $\mathrm{u}^*(b)$ to within a user specified tolerance.

For expositional simplicity, we consider a boundary value problem with one constraint located at each boundary of the domain, namely, $\big(\mathrm{v}(a), \mathrm{u}(b)\big) = \big(\mathrm{v}^*(a), \mathrm{u}^*(b)\big)$. We treat the boundary value, $\mathrm{u}^*(b)$, as a data point and consider the solution as an inference problem over the unspecified initial value, $\mathrm{u}(a)$. The likelihood therefore defines the mismatch between the boundary value $\mathrm{u}(b)$, obtained from the realised probabilistic solution given some $\mathrm{u}(a)$, with the exact boundary value, $\mathrm{u}^*(b)$, as follows,

$$p(\mathrm{u}^*(b) \mid \mathrm{u}(a), \mathrm{v}^*(a), \theta, \Psi, N) = \mathcal{N}\big(\mathrm{u}^*(b) \mid \mathrm{m}^{(\mathrm{u})}(b), \mathrm{C}^{(\mathrm{u})}(b, b)\big). \tag{12}$$

where $\mathrm{m}^{(\mathrm{u})}(b)$ and $\mathrm{C}^{(\mathrm{u})}(b, b)$ are the posterior mean and covariance for state u at time point b obtained from evaluating Algorithm 1. The posterior distribution of the states therefore has density,

$$\begin{aligned} & p\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t}), \mathrm{u}(a) \mid \mathrm{u}^*(b), \mathrm{v}^*(a), \theta, \Psi, N\big) \\ & \propto p\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t}) \mid \mathrm{u}(a), \mathrm{u}^*(b), \mathrm{v}^*(a), \theta, \Psi, N\big)\; p(\mathrm{u}^*(b) \mid \mathrm{u}(a), \mathrm{v}^*(a), \theta, \Psi, N)\; p\big(\mathrm{u}(a)\big), \end{aligned} \tag{13}$$

which exhibits multimodality over the states, as shown on the left side of Figure 4. While deterministic numerical solvers rely on an ad hoc end point mismatch tolerance, the probabilistic framework naturally

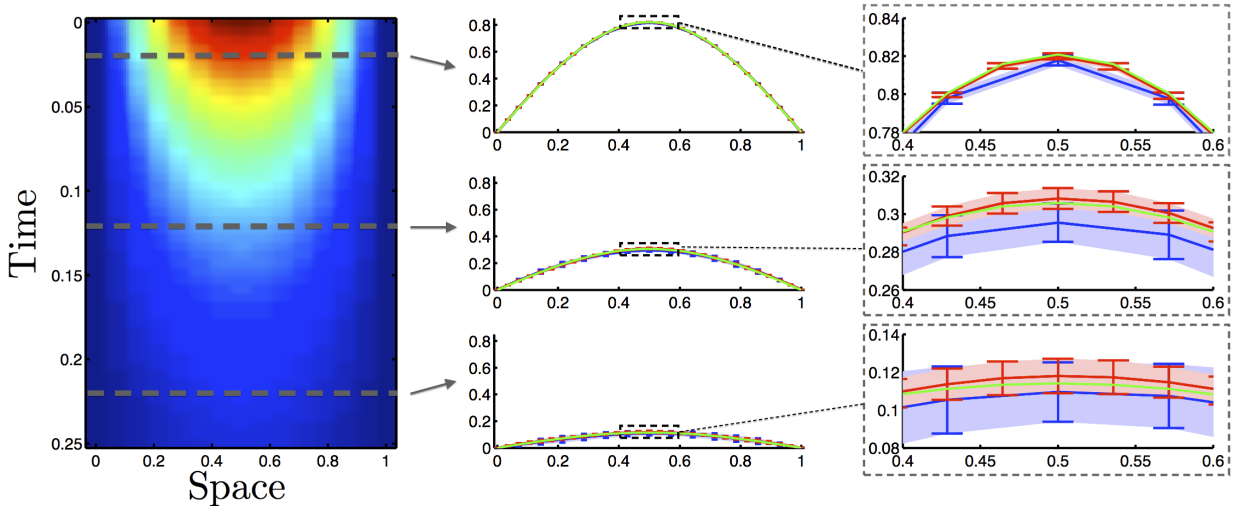

Figure 2: We illustrate the probabilistic output of the solution to the heat equation PDE, with $\kappa = 1$, integrated between $t = 0$ and $t = 0.25$ using two grid sizes; the coarser mesh (shown in blue) consists of 15 spatial discretisation points and 50 time discretisation points, the finer mesh consists of 29 spatial discretisation points and 100 time discretisation points. We show the spatial posterior predictions at three time points; $t = 0.02$ (top), $t = 0.12$ (middle) and $t = 0.22$ (bottom). The exact solution at each time point is represented by the green line. The error bars show the mean and 2 standard deviations for each of the probabilistic solutions calculated using 50 simulations.

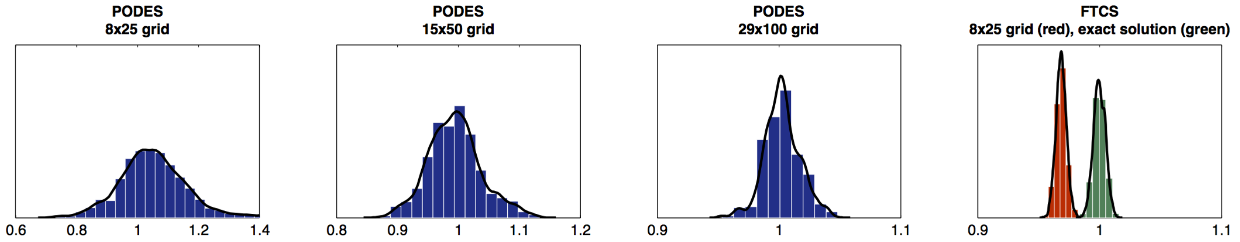

Figure 3: We illustrate the inverse problem by performing inference over the parameter κ in the heat equation, integrated between $t = 0$ and $t = 0.25$. We generate data over a grid of 8 spatial discretisation points and 25 time discretisation points by using the exact solution with $\kappa = 1$, then adding noise with standard deviation of 0.005. We firstly use the probabilistic differential equation solver (PODES) using three grid sizes; a coarse mesh consisting of 8 spatial discretisation points and 25 time discretisation points (far left), a finer mesh consisting of 15 spatial discretisation points and 50 time discretisation points (second from left), and a further finer mesh consisting of 29 spatial discretisation points and 100 time discretisation points (second from right). Note the change in scale as the posterior variance decreases with increasing resolution of the discretisation. As an illustrative comparison, we show the posterior distributions using a deterministic forward in time, centred in space (FTCS) integration scheme (far right). If the discretisation is not fine enough, we obtain an overconfident biased posterior that assigns *negligible* probability mass to the true value of κ. In contrast, use of the exact solution produces a perfectly unbiased posterior, as expected.

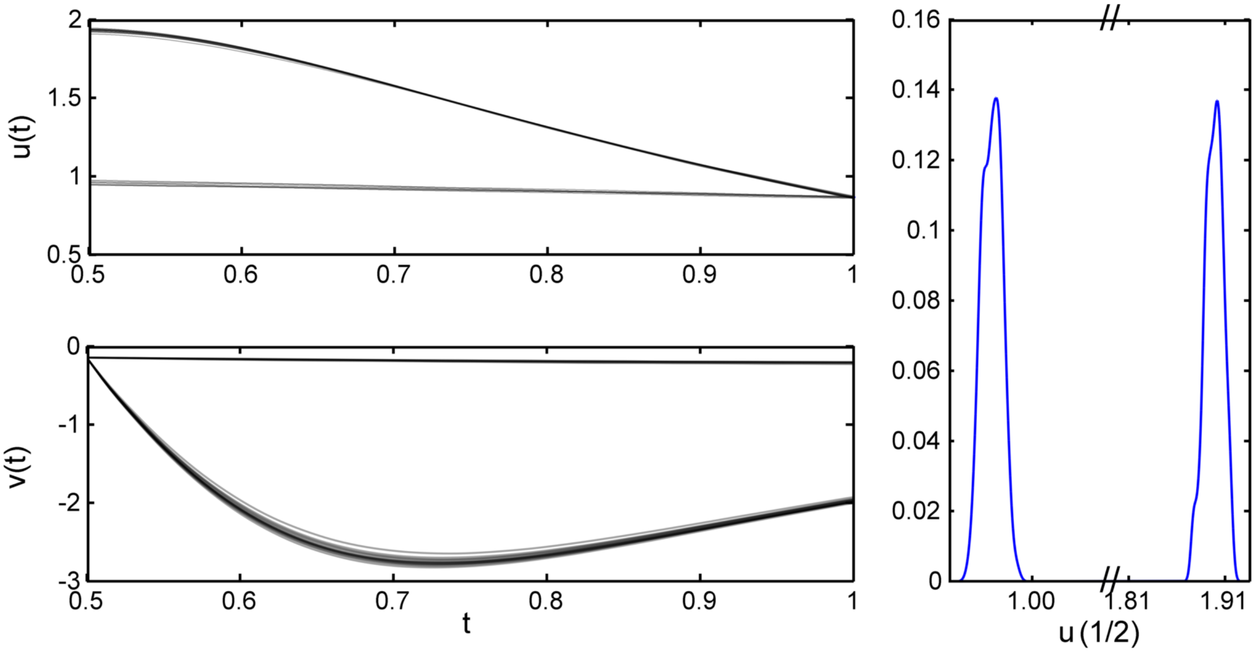

Figure 4: 3,000 samples were drawn from the posterior probability density (13) and the trajectories are shown for both states (left, above and below). The marginal posterior probability density over the unknown initial condition $u(\frac{1}{2})$ is also shown (right).

defines that tolerance through the predictive distribution of (12). An important consideration for solving MBVPs probabilistically is to sample efficiently from potentially multimodal posteriors over the initial value. We therefore recommend the use of an appropriate MCMC scheme such as parallel tempering (Geyer, 1991), which can quickly identify and explore disjoint regions of high posterior probability. We provide one such implementation of parallel tempering in the Appendix (Algorithm 5).

As a demonstration of probabilistically solving a mixed boundary value problem, we consider a special case the Lane-Emden model, which is used to describe the density u of gaseous spherical objects, such as stars, as a function of its radius t, (Shampine, 2003). We rewrite the canonical second order ODE as a system of a first order equations with a mixed boundary value,

$$\left\{\begin{array}{lll} \mathrm{u}_t(t) & = \mathrm{v}(t), & t \in [\frac{1}{2}, 1], \\ \mathrm{v}_t(t) & = -2\mathrm{v}(t)/t - \mathrm{u}^5(t), & t \in [\frac{1}{2}, 1], \\ \big(\mathrm{u}(1), \mathrm{v}(\frac{1}{2})\big) & = \big(\mathrm{u}^*(1), \mathrm{v}^*(\frac{1}{2})\big), & \end{array}\right. \tag{14}$$

with boundary conditions, $\mathrm{u}^*(1) = \sqrt{3}/2$ and $\mathrm{v}^*(\frac{1}{2}) = -288/2197$.

The unknown initial state $\mathrm{u}^*(\frac{1}{2})$ is assigned a diffuse Gaussian prior with mean 1.5 and standard deviation $2\big|\mathrm{u}^*(1) - \mathrm{v}^*(\frac{1}{2})\big|$, which reflects the possibility that multiple solutions may be present over a wide range of initial states. In this example, we chose the squared exponential covariance to model what we expect to be a very smooth solution. The discretisation grid consists of 100 equally spaced points. The length-scale is set to twice the discretisation grid step size and the prior precision is set to 1. Figure 4 shows a posterior sample from Equation 13 and identifies two high density regions in the posterior corresponding to distinct trajectories that approximately satisfy model dynamics given our discretisation grid. Figure 4 also shows multimodality through the marginal posterior over the unknown initial state $\mathrm{u}(\frac{1}{2})$.

This example illustrates the need for accurately modelling solution uncertainty in a functional manner. A numerical approximation, even with corresponding numerical error bounds, will fail to detect the existence of a second solution, whereas a probabilistic approach allows us to determine the number and location of possible solutions. In this case, a probabilistic approach to solving the differential equation problem addresses model bias caused by multiplicity of solutions.

6.3 Forward Simulation for a PDE Model of Fluid Dynamics

We present the following example as a proof of concept that the probabilistic solver may be applied reliably and straightforwardly to a very high-dimensional dynamical system. The Navier-Stokes system is a fundamental model of fluid dynamics, incorporating laws of conservation of mass, energy and linear momentum, as well as physical properties of an incompressible fluid over some domain given constraints imposed along the boundaries. It is an important component of complex models in oceanography, weather, atmospheric pollution, and glacier movement. Despite its extensive use, the dynamics of Navier-Stokes models are poorly understood even at small time scales, where they can give rise to turbulence.

We consider the Navier-Stokes PDE model for the time evolution of 2 components of the velocity, $\mathrm{u} : \mathcal{D} \to \mathbb{R}^2$, of an incompressible fluid on a torus, $\mathcal{D} := [0, 2\pi,) \times [0, 2\pi]$, expressed in spherical coordinates. The Navier-Stokes boundary value problem is defined by:

$$\begin{cases} \mathrm{u}_t - \theta\, \Delta \mathrm{u} + \big(\mathrm{u} \cdot \nabla\big)\mathrm{u} & = & \mathsf{f} - \nabla \mathsf{p}, & (x,t) \in \mathcal{D} \times [a,b], \\ \nabla \cdot \mathrm{u} & = & 0, & (x,t) \in \mathcal{D} \times [a,b], \\ \int \mathrm{u}^{(j)}\, \mathrm{d}x & = & 0, & (x,t) \in \mathcal{D} \times [a,b],\ j = 1,2, \\ \mathrm{u} & = & \mathrm{u}^*, & (x,t) \in \mathcal{D} \times \{0\}, \end{cases} \tag{15}$$

where $\Delta := \left[\frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2}\right]$ is the Laplacian operator such that $\Delta \mathrm{u} = \big(\mathrm{u}^{(1)}_{x_1 x_1} + \mathrm{u}^{(1)}_{x_2 x_2}, \mathrm{u}^{(2)}_{x_1 x_1} + \mathrm{u}^{(2)}_{x_2 x_2}\big)$, and $\nabla := \left[\frac{\partial}{\partial x_1} + \frac{\partial}{\partial x_2}\right]$ is the gradient operator such that $\nabla \mathrm{u} = \big(\mathrm{u}^{(1)}_{x_1} + \mathrm{u}^{(1)}_{x_2}, \mathrm{u}^{(2)}_{x_1} + \mathrm{u}^{(2)}_{x_2}\big)$. The model is parameterised by the viscosity of the fluid, $\theta \in \mathbb{R}^+$, the pressure function $\mathsf{p} : \mathcal{D} \times [a,b] \to \mathbb{R}$, and the external time-homogeneous forcing function $\mathsf{f} := 5 \times 10^{-3} \cos\left[\left(\frac{1}{2}, \frac{1}{2}\right) \cdot x\right]$. We further assume periodic boundary conditions, and viscosity $\theta = 10^{-4}$ in the turbulent regime. The exact solution of the Navier-Stokes boundary value problem (15) is not known in closed form.

Often, the quantity of interest is the local spinning motion of the incompressible fluid, called vorticity, which we define as,

$$\varpi = -\nabla \times \mathrm{u},$$

where $\nabla \times \mathrm{u}$ represents the rotational curl defined as the cross product of ∇ and u, with positive vorticity corresponding to clockwise rotation. This variable will be used to better visualise the probabilistic solution of the Navier-Stokes system by reducing the two components of velocity to a one dimensional function. We discretize the Navier-Stokes model (15) over a grid of size 128 in each spatial dimension. Therefore, a pseudo spectral projection in Fourier space yields 16,384 coupled, stiff ODEs with associated constraints. Full details of the pseudo spectral projection are provided to allow full replication of these results and are available on the accompanying website. Figure 6.3 shows four forward simulated vorticity trajectories (along rows), obtained from two components of velocity governed by the Navier-Stokes equations (15) at four distinct time points (along columns). Slight differences in the state dynamics can be seen at the last time point, where the four trajectories visibly diverge from one another. These differences express the epistemic uncertainty resulting from discretising the exact but unknown infinite dimensional solution.

6.4 Forward Simulation for a DIFP Model of Cellular Biochemical Dynamics

The probabilistic approach for forward simulation of DIFPs is described in Algorithm 6 in the Appendix. In addition to modelling discretisation uncertainty in a structured, functional way, our framework allows us to straightforwardly incorporate uncertainty in the initial function through the forward simulation; indeed, the initial function ϕ may iteself only be available at a finite number of nodes. The probabilistic approach quantifies the uncertainty associated with the estimation of $\phi(t)$ and propagates it recursively through the states. Even when $\phi(t)$ is fully specified in advance, the dependence of the current state on previously estimated states impacts the accuracy of the numerical solution using standard solvers, even over the short term. In the following example we account for the uncertainty associated with current and delayed estimates of system states, and fully address these potential sources of bias in the estimation of model parameters.

The JAK-STAT mechanism describes a series of reversible biochemical reactions of STAT-5 transcription factors in response to binding of the Erythropoietin (Epo) hormone to cell surface receptors (Pellegrini and Dusanter-Fourt, 1997). After gene activation occurs within the nucleus, the transcription factors revert to

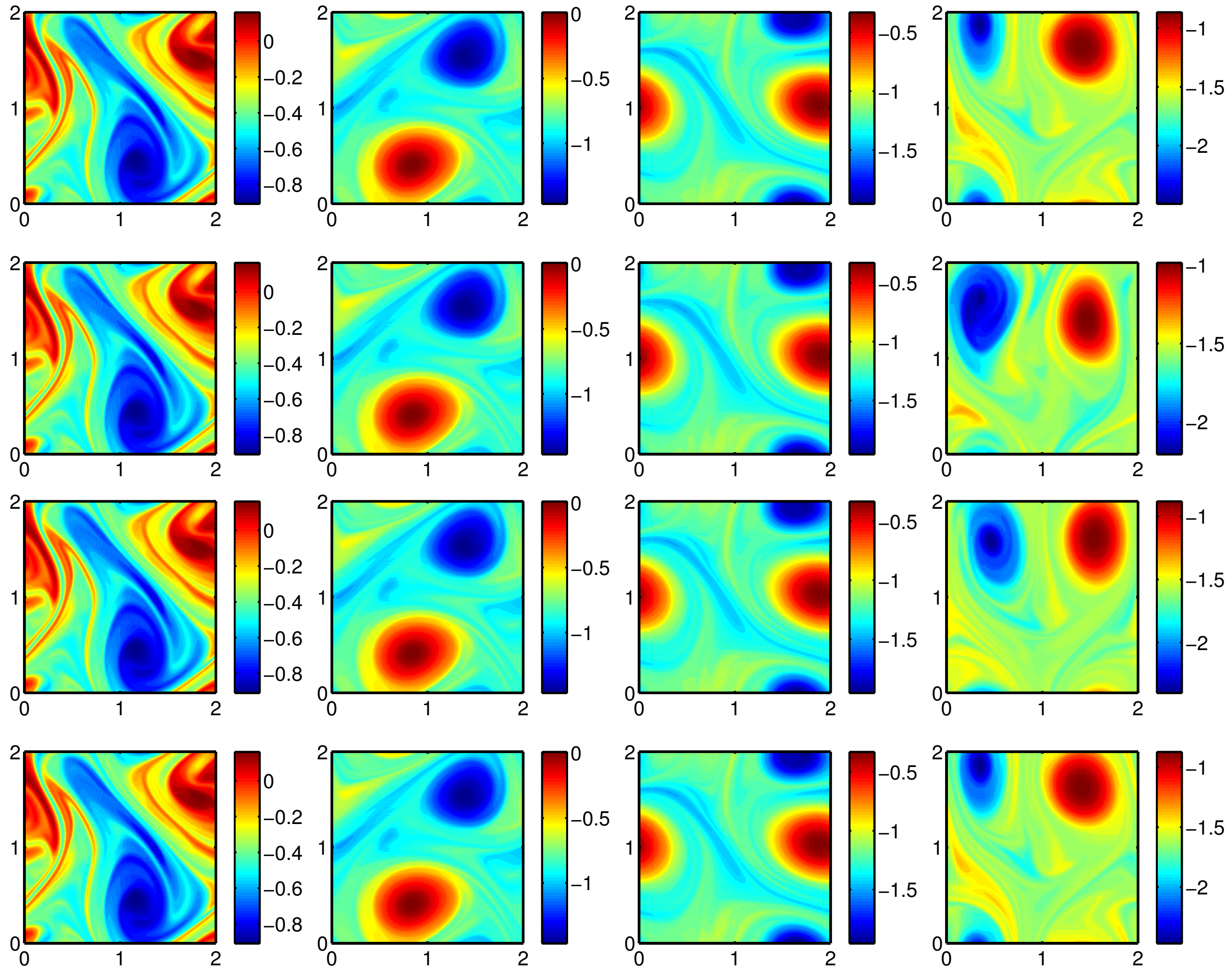

Figure 5: Time evolution of four forward simulated realisations (along rows) of fluid vorticity, governed by the forced Navier-Stokes model, over two spatial dimensions: the angle of the inner ring (horizontal axis) and outer ring (vertical axis) of a two dimensional torus. Angles are expressed in radians. Vorticities are evaluated at times $\mathbf{t} = [0.2, 0.4, 0.6, 0.8]$ units (along columns).

their initial state, returning to the cytoplasm to be used in the next activation cycle. This last stage is not well understood and is proxied in the model by the unknown time delay τ. The model for this mechanism describes changes in 4 reaction states of STAT-5 through the nonlinear DIFP,

$$\begin{cases} \mathrm{u}_t^{(1)}(t) & = - & k_1\, \mathrm{u}^{(1)}(t)\ \mathrm{Epo}R_A(t) + \ 2\, k_4\, \mathrm{u}^{(4)}(t-\tau), & t \in [0, 60], \\ \mathrm{u}_t^{(2)}(t) & = & k_1\, \mathrm{u}^{(1)}(t)\ \mathrm{Epo}R_A(t) - k_2\, {\mathrm{u}^{(2)}}^2(t), & t \in [0, 60], \\ \mathrm{u}_t^{(3)}(t) & = - & k_3\, \mathrm{u}^{(3)}(t) + \ \frac{1}{2}\, k_2\, {\mathrm{u}^{(2)}}^2(t), & t \in [0, 60], \\ \mathrm{u}_t^{(4)}(t) & = & k_3\, \mathrm{u}^{(3)}(t) - k_4\, \mathrm{u}^{(4)}(t-\tau), & t \in [0, 60], \\ \mathrm{u}(t) & = & \phi(t), & t \in [-\tau, 0]. \end{cases} \tag{16}$$

The initial function components $\phi^{(2)}(t) = \phi^{(3)}(t) = \phi^{(4)}(t)$ are everywhere zero, while the constant initial function $\phi^{(1)}(t)$ is unknown.

6.4.1 Indirectly Observed Measurements

The states for this system cannot be measured directly, but are observed through a nonlinear transformation, $\mathcal{G} : \mathbb{R}^3 \times \Theta \to \mathbb{R}^4$, defined as,

$$\mathcal{G}(\mathrm{u}, \theta) = \begin{bmatrix} k_5 \left(\mathrm{u}^{(2)} + 2\mathrm{u}^{(3)}\right) \\ k_6 \left(\mathrm{u}^{(1)} + \mathrm{u}^{(2)} + 2\mathrm{u}^{(3)}\right) \\ \mathrm{u}^{(1)} \\ \mathrm{u}^{(3)} / \left(\mathrm{u}^{(2)} + \mathrm{u}^{(3)}\right) \end{bmatrix}, \tag{17}$$

and parameterised by the unknown scaling factors k_5 and k_6. The indirect measurements of the states are assumed contaminated with additive zero-mean Gaussian noise, $\varepsilon(\mathbf{t})$, with experimentally determined standard deviations,

$$y(\mathbf{t}) = \mathcal{G}\big(\mathrm{u}(\mathbf{t}), \theta\big) + \varepsilon(\mathbf{t}).$$

Our analysis is based on experimental data measured at the locations, $\mathbf{t}$, from Swameye et al. (2003), which consists of 16 measurements for the first two states of the observation process. Raue et al. (2009) further utilise an additional artificial data point for each of the third and fourth observation process states to deal with lack of parameter identifiability for this system; we therefore adopt this assumption in our analysis. The forcing function, $\mathrm{Epo}R_A : [0, 60] \to \mathbb{R}^+$, is not known, but measured at 16 discrete time points $\mathbf{t}_{EPO}$. As per Raue et al. (2009), we assume that observations are measured without error. We further assume that this function shares the same smoothness as the solution state (piecewise linear first derivative). The full conditional distribution of the forcing function is given by a GP interpolation of the observations. Forward inference for this model will be used within the statistical inverse problem of recovering unknown parameters and first initial state, $\theta = [k_1, \ldots, k_6, \tau, \mathrm{u}^{(1)}(0)]$, from experimental data.

6.4.2 Inference of Unknown Model Parameters

We demonstrate fully probabilistic inference for state trajectories and parameters of the challenging 4 state delay initial function model (16) describing the dynamics of the JAK-STAT cellular signal transduction pathway (Raue et al., 2009). There have been several analyses of the JAK-STAT pathway mechanism based on this data (e.g., Campbell and Chkrebtii, 2013; Raue et al., 2009; Schmidl et al., 2003; Swameye et al., 2003). Despite the variety of modelling assumptions considered by different authors, as well as distinct inference approaches, some interesting common features have been identified that motivate the explicit modelling of discretisation uncertainty for this application. Firstly, the inaccuracy and computational constraints of numerical techniques required to solve the system equations have led some authors to resort to coarse ODE approximations or even indirectly bypassing numerical solution via Generalised Smoothing. This motivates a formal analysis of the structure and propagation of discretisation error through the inverse problem. A further issue is that the model (16) and its variants suffer from model misspecification. The above studies suggest the model is not flexible enough to fit the available data, however it is not clear how much of this misfit is due to model discrepancy, and how much may be attributed to discretisation error.

Our analysis proceeds by defining prior distributions on the unknown parameters as follows,

$$
\begin{aligned}
k_i &\sim \text{Exp}\,(1) \quad i = 1, \dots, 6 \\
\tau &\sim \chi^2_6 \\
\mathrm{u}^{(1)}(0) &\sim \mathcal{N}\big(y^{(3)}(0), 40^2\big) \\
\alpha + 100 &\sim \text{Log-}\mathcal{N}\,(10, 1) \\
\lambda &\sim \chi^2_1
\end{aligned}
$$

We obtained samples from the posterior distribution of the model parameters, $\theta = [k_1, \dots, k_6, \tau, \mathrm{u}^{(1)}(0)]$, solution states, $\mathrm{u}(\mathbf{t}, \theta)$, and auxiliary variables, Ψ given the data $y(\mathbf{t})$ using MCMC. In order to construct a Markov chain that efficiently traverses the parameter space of this multimodal posterior distribution, we employed a parallel tempering sampler (Geyer, 1991) with 10 parallel chains along a uniformly spaced temperature profile over the interval $[0.5, 1]$. Each probabilistic DIFP simulation was generated using Algorithm 6 under an equally spaced discretisation grid of size $N = 500$. Full algorithmic details are provided in the Appendix (Algorithm 8).

We obtained two groups of posterior samples, each of size 50,000. Within chain convergence was assessed by testing for equality of means between disjoint iteration intervals of the chain (Geweke 1992). Between chain convergence was similarly assessed. Additionally we ensured that the acceptance rate fell roughly within the accepted range of 18%–28% for each of the two parameter blocks, and that the total acceptance rate for moves between any two chains remained roughly within 5%-15%.

6.4.3 Posterior Inference Results

Correlation plots and kernel density estimates with priors for the marginal parameter posteriors are shown in Figure 6. All parameters with the exception of the prior precision α are identified by the data, including the rate parameter k_2 which appears to be only weakly identified. We observe strong correlations between parameters, consistent with previous studies on this system. For example, there is strong correlation among the scaling parameters k_5, k_6 and the initial first state $\mathrm{u}^{(1)}(0)$. Interestingly, there appears to be a correlation between the probabilistic solver's length-scale λ and the first, third and fourth reaction rates. Furthermore, this correlation has a nonlinear structure, where the highest parameter density region seems to change with length scale implying strong sensitivity to the solver specifications. The length-scale is the probabilistic analogue, under a bounded covariance, of the step number in a numerical method. However, in analyses based on numerical integration, the choice of a numerical technique effectively fixes this parameter at an arbitrary value that is chosen a priori. Our result here suggests that for this problem, the inferred parameter values are highly and nonlinearly dependent on the choice of the numerical method used. We speculate that this effect may become quite serious for more sensitive systems, such that ignoring discretisation uncertainty within the inverse problem may result in inferential bias.

A sample from the marginal posterior of state trajectories and the corresponding observation process are shown in Figure 7. The error bars on the data points show two standard deviations of the measurement error from Swameye et al. (2003). It is immediately clear that our model, which incorporates discretisation uncertainty in the forward problem, still does not fully capture the dynamics of the observed data. This systematic lack of fit suggests the existence of model discrepancy beyond that described by discretisation uncertainty.

7 Discussion

This paper has presented a probabilistic formalism to describe the structure of approximate solution uncertainty for general systems of differential equations. Rather than providing a single set of discrete function values that approximately satisfy the constraints imposed by the system of differential equations, our approach yields a probability measure over the space of such infinite dimensional functions. This is a departure

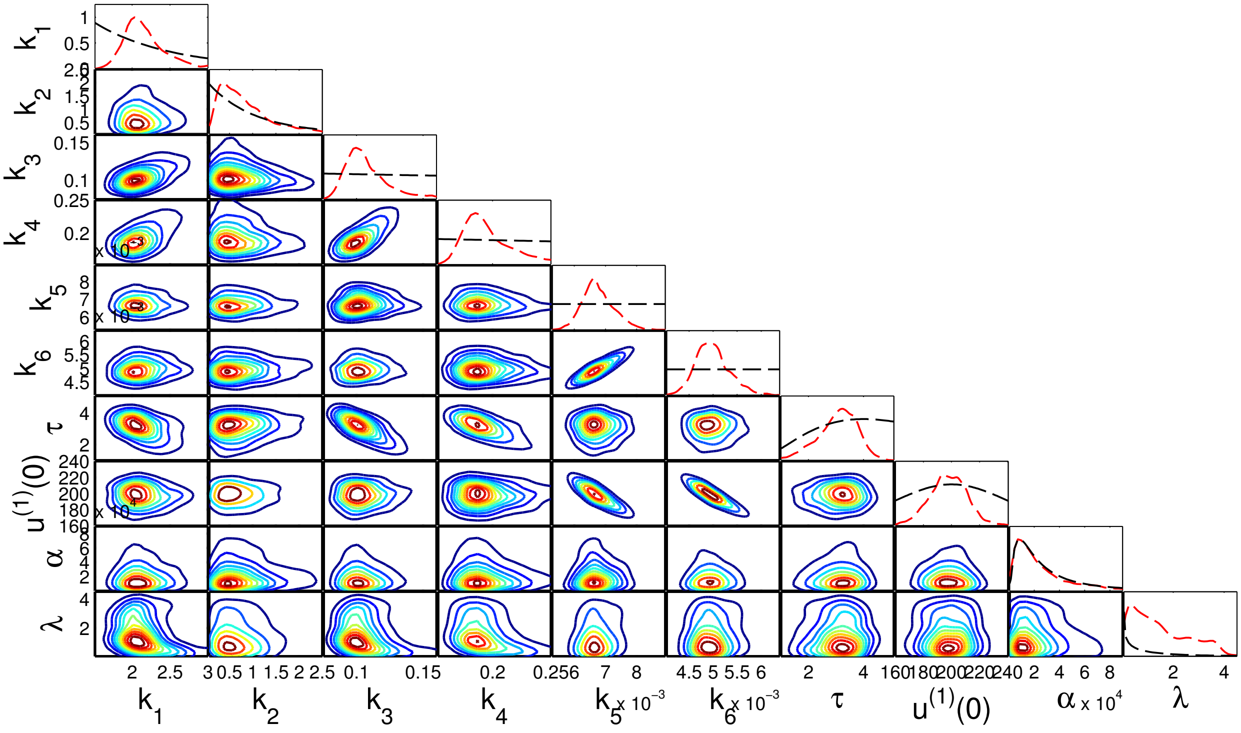

Figure 6: Marginal posterior distribution of the model parameters using probabilistic forward simulation based on a sample of size 50,000, generated using a parallel tempering algorithm with ten chains. Prior probability densities are shown in black.

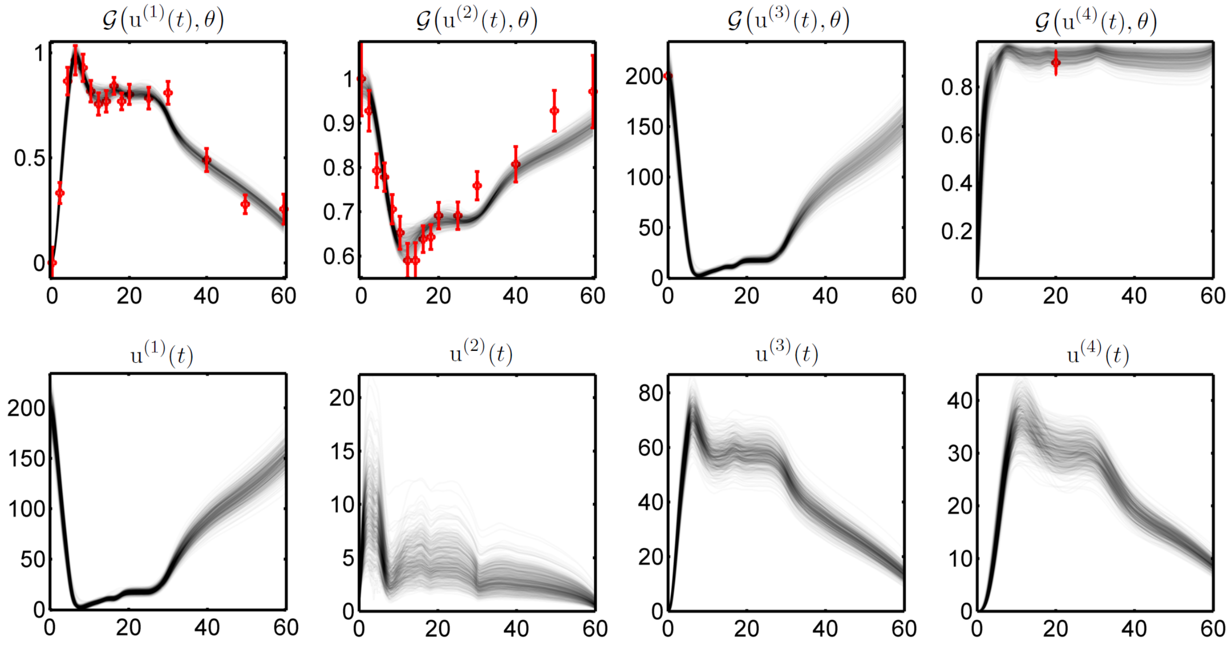

Figure 7: Experimental data (red points) and error bars showing two standard deviations of the measurement error from Swameye et al. (2003). Top row: sample paths (grey lines) of the observation processes obtained by transforming a sample from the marginal posterior distribution of the states (bottom row) by the observation function (17).

from the existing accepted practice of employing a deterministic numerical integration code to obtain an approximate statistical error model as part of the inference process. This enables adoption of probability calculus for coherently propagating functional uncertainty when solving the statistical inverse problem.

7.1 Practical Implementations and Efficiency

We have demonstrated that our probabilistic framework can already be applied to complex and high dimensional systems using the Navier-Stokes example. However, further work is needed to improve computational efficiency to a point where it is comparable with existing implementations of numerical integration codes. In principle, as the probabilistic integration method we have proposed relies on solutions of linear systems, it shares the same algorithmic scaling as most implicit numerical integration codes (e.g. Crank-Nicholson), and it is anticipated that, over time, algorithmic and code development will ensure our methodology attains similar levels of performance, becoming part of the standard toolbox for climate researchers, geoscientists, and engineers. Algorithmic advances and code development will be an exciting and fruitful area of ongoing investigation with high impact.

Extension of the methodology to systems that exhibit directional derivative errors is immediate, for example in fluid dynamics applications. Indeed, in Section 4 we have modelled the mismatch between the prior on the derivative and the ODE model as Gaussian. Relaxing this assumption no longer guarantees a closed form representation of the updated prior on the state derivative, however this can be overcome by an additional layer of Monte Carlo sampling within Algorithm 1. Such questions would additionally motivate the development of efficient MCMC algorithms in this context.

In this work we have been able to directly exploit the structure of the mathematical model within the inference process in a more informed manner than by treating the intractable system of differential equations and associated numerical solver code as a “black box”. We may also exploit the intrinsic manifold structure induced by the probabilistic model posterior to increase efficiency of a variety of MCMC methods (Girolami and Calderhead, 2011). As we have already noted, controlling the auxiliary parameters Ψ associated with smoothness of the space of possible trajectories can also add flexibility to MCMC sampling methods. This feature of our probabilistic method allows one to define intermediate target densities for population MCMC methods such as Smooth Functional Tempering (Campbell and Steele, 2011), thereby increasing efficiency when exploring the complex posterior densities arising in applications described by systems of nonlinear ordinary or partial differential equations.

The proposed probabilistic approach takes the form of a linear functional projection, allowing direct computation of sensitivities of the system states with respect to parameters. Local sensitivity analysis can be useful in engineering applications, inference, or in aiding optimisation. This would also allow, for example, our probabilistic framework to be incorporated into a classical nonlinear least squares framework (Bates and Watts, 2007).

7.2 Relationship and Connections to Other Areas

Fundamentally, the probabilistic approach we advocate allows one to define a formal tradeoff between uncertainty induced by numerical accuracy and the size of the inverse problem, by choice of discretisation grid. This problem is of deep interest in the uncertainty quantification community (see, for example, Arridge et al., 2006; Kaipio et al., 2004). We may now quantify and compare the individual contributions to overall uncertainty from discretisation, model misspecification, and Monte Carlo error within the probabilistic framework.

Our probabilistic model of discretisation uncertainty may be used to guide mesh refinement and indeed mesh design for complex models, in that a predictive distribution is now available, which forms the basis of experiment design approaches to mesh development. Numerical solutions of ODEs and PDEs for complex models also rely on adaptive mesh selection. The probabilistic formalism presented here can now inform mesh selection in a probabilistic way, by optimising a chosen design criterion. Some work in this direction has already been conducted in Chkrebtii (2013), where the natural Kullback-Leibler divergence criterion is successfully used to adaptively choose the discretisation grid probabilistically. Such an approach was also suggested by Skilling (1991) using the cross-entropy.

Inference and prediction for computer experiments (Sacks et al., 1989) relies on numerical solutions of large-scale system models. Currently, numerical uncertainty in the model is largely ignored, although it is informally incorporated through a covariance nugget in the emulator (see, for example, Gramacy and Lee, 2012). Adopting a probabilistic approach on a large scale will have practical implications in this area by permitting relaxation of the error-free assumption adopted when modelling computer code output, leading to more realistic and flexible emulators.

In modelling uncertainty about an exact but unknown solution, the probabilistic integration formalism may be viewed as providing both an estimate of the exact solution and its associated functional error analysis. Indeed, many existing numerical methods can be interpreted in the context of this general probabilistic framework. This suggests the possibility to both generalise existing numerical solvers and to develop new sampling schemes to complement the probabilistic solutions we describe in this contribution.

7.3 Chaotic Systems

Chaotic systems arise in modelling a large variety of physical systems, including laser cavities, chemical reactions, fluid motion, crystal growth, weather prediction, and earthquake dynamics (Baker and Gollub, 1996, chapter 7). Extreme sensitivity to small perturbations characterises chaotic dynamics, where the effect of discretisation uncertainty becomes an important contribution to global solution uncertainty. Although the long term behaviour of the system is entrenched in the initial states, the presence of numerical discretisation error results in exponential divergence from the exact solution. Consequently, information about initial states rapidly decays as system solution evolves in time. This insight, demonstrated and explained by (Berliner, 1991), is showcased in the following example. We consider the classical Lorenz initial value problem (Lorenz, 1963), a deceptively simple three-state ODE model of convective fluid motion induced by a temperature difference between an upper and lower surface. A sample from the probabilistic posterior of this system is shown in Figure 8 given model parameters in the chaotic regime and a fixed initial state. There is a short time window within which there is negligible uncertainty in the solution, but the accumulation of uncertainty quickly results in divergent, yet highly structured, solutions as the flow is restricted around a chaotic attractor. This example highlights the need for a functional model of discretization uncertainty to replace numerical pointwise error bounds. Within the inverse problem, a probabilistic approach also allows us to quantify the loss of model information as we move away from the initial state.

The study of chaotic systems is clearly a very important area of research, and it is anticipated that the probabilistic approach proposed may help to accelerate progress.

7.4 Outlook

We now return to an important discussion regarding the class of problems for which probabilistic integration is well suited. As we have seen with inference for the JAK-STAT model, even relatively small discretisation uncertainty can become amplified through the nonlinear forward model and may introduce bias in the estimation of parameters to which the trajectory is highly sensitive. Nevertheless, it may be conjectured that inference for low dimensional, stable systems, without any topological restrictions on the solution may not suffer from significant discretisation effects. However, the benefits of our formulation undoubtedly lie at the frontier of research in uncertainty quantification, dealing with massive nonlinear models, exhibiting complex or even chaotic dynamics, and strong spatial-geometric effects (e.g. subsurface flow models). Indeed, solving such systems approximately is a problem that is still at the edge of current research in numerical analysis. We suggest that a probabilistic approach would provide an important additional set of tools for the study of such systems.

In conclusion, we hope that our paper will encourage much more research at this exciting interface between mathematical analysis and statistical methodology.

Acknowledgements

We would like to thank Tan Bui, Simon Byrne, Mike Dowd, Des Higham, Richard Lockhart, Youssef Marzouk, Giovanni Petris, Daniel Simpson, John Skilling, and Andrew Stuart for their invaluable comments and

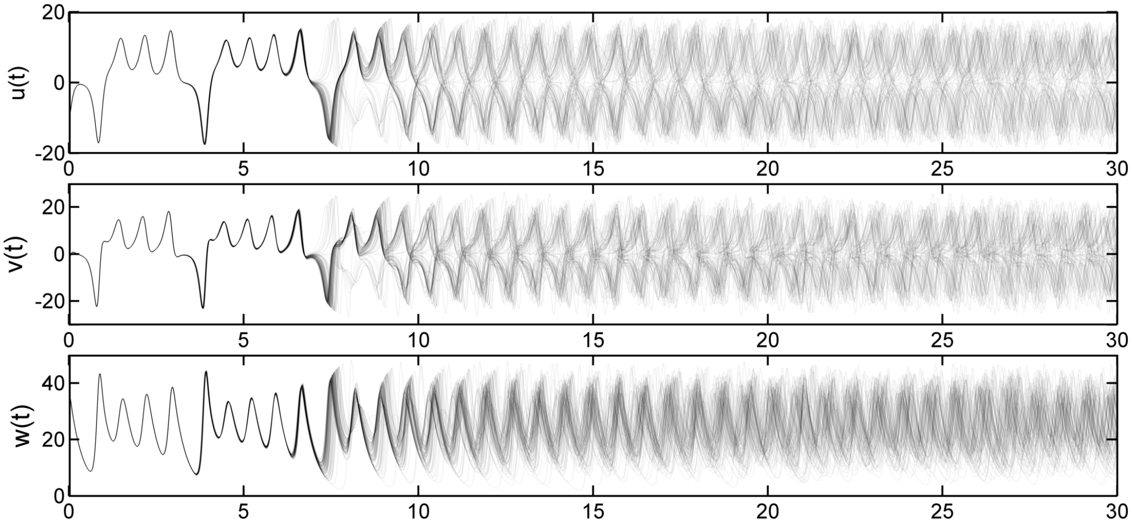

Figure 8: One hundred samples from a probabilistic solution posterior process for the Lorenz system (Lorenz, 1963) under fixed initial state, $(-11, -5, 38)$. Probabilistic trajectories for the Lorenz system were obtained using our proposed methodology, given model parameters $\theta = (10, 8/3, 28)$, in the chaotic regime. The probabilistic solution is obtained using a discretization grid of 3000 equally spaced points on the interval $(0, 30)$. The squared exponential covariance function is chosen for this application based on our assumption of a locally smooth solution. We set the length scale λ equal to twice the step size, which allows us to assign the largest weight to data generated at the last available solver knot. The prior precision α is set to the low value of 10^{-3}, reflecting our prior knowledge that the system exhibits chaotic dynamics.

suggestions. In addition we are greatly indebted to the reviewers for their constructive comments which helped to improve the quality of the paper immensely.

We would also like to acknowledge the funding agencies that have supported this research. O. Chkrebtii and D. A. Campbell are supported by the Natural Sciences and Engineering Research Council of Canada. M.A. Girolami is supported by the UK Engineering and Physical Sciences Council (EPSRC) via the Established Career Research Fellowship EP/J016934/1 and the Programme Grant Enabling Quantification of Uncertainty for Large-Scale Inverse Problems, EP/K034154/1, `http://www.warwick.ac.uk/equip`. He also gratefully acknowledges support from a Royal Society Wolfson Research Merit Award. B. Calderhead gratefully acknowledges his previous Research Fellowship through the 2020 Science programme, funded by EPSRC grant number EP/I017909/1 and supported by Microsoft Research, as well as support in part by EPSRC Mathematics Platform grant EP/I019111/1.

References

Arridge, S. R., J. P. Kaipio, V. Kolehmainen, M. Schweiger, E. Somersalo, T. Tarvainen, and M. Vauhkonen (2006). Approximation errors and model reduction with an application in optical diffusion tomography. *Inverse Problems 22*(1), 175–195.

Ascher, U. M., R. M. Mattheij, and R. D. Russell (1988). *Numerical solution of boundary value problems for ordinary differential equations.* Prentice-Hall.

Baker, G. and J. Gollub (1996). *Chaotic Dynamics: an Introduction.* Cambridge University Press.

Bates, D. and D. Watts (2007). *Nonlinear regression analysis and its applications.* Wiley series in probability and statistics. Probability and statistics section. John Wiley & Sons.

Bellen, A. and M. Zennaro (2003). *Numerical Methods for Delay Differential Equations.* Clarendon Press.

Berliner, L. M. (1991). Likelihood and Bayesian prediction of chaotic systems. *Journal of the American Statistical Association 86*(416), 938–952.

Beyn, W. and E. Doedel (1981). Stability and multiplicity of solutions to discretizations of nonlinear ordinary differential equations. *SIAM Journal on Scientific and Statistical Computing 2*(1), 107–120.

Bock, H. G. (1983). Recent advances in parameter identification techniques for ode. In P. Deuflhard and E. Harrier (Eds.), *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, pp. 95–121. Birkhuser.

Brunel, N. J. (2008). Parameter estimation of ODEs via nonparametric estimators. *Electronic Journal of Statistics 2*, 1242–1267.

Brynjarsdóttir, J. and A. O'Hagan (2014). Learning about physical parameters: The importance of model discrepancy. Submitted.

Butcher, J. (2008). *Numerical Methods for Ordinary Differential Equations.* John Wiley and Sons Ltd.

Calderhead, B. and M. Girolami (2011). Statistical analysis of nonlinear dynamical systems using differential geometric sampling methods. *Interface Focus 1*(6), 821–835.

Calderhead, B., M. Girolami, and N. Lawrence (2009). Accelerating Bayesian inference over nonlinear differential equations with gaussian processes. *Advances in Neural Information Processing Systems 21*, 219–224.

Campbell, D. and S. Lele (2013). An ANOVA test for parameter estimability using data cloning with application to statistical inference for dynamic systems. *Computational Statistics and Data Analysis*.

Campbell, D. and R. J. Steele (2011). Smooth functional tempering for nonlinear differential equation models. *Statistics and Computing 22*, 429–443.

Campbell, D. A. and O. Chkrebtii (2013). Maximum profile likelihood estimation of differential equation parameters through model based smoothing state estimates. *Mathematical Biosciences 246*, 283–292.

Chkrebtii, O. (2013). *Probabilistic solution of differential equations for Bayesian uncertainty quantification and inference.* Ph. D. thesis, Simon Fraser University.

Coddington, A. and N. Levinson (1955). *Theory of ordinary differential equations.* International series in pure and applied mathematics. McGraw-Hill.

Conti, S. and A. O'Hagan (2010). Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference 140*, 640–651.

Diaconis, P. (1988). *Bayesian numerical analysis.* Springer-Verlag.

Dowd, M. (2007). Bayesian statistical data assimilation for ecosystem models using Markov Chain Monte Carlo. *Journal of Marine Systems 68*(3-4), 439–456.

Geyer, C. (1991). Markov chain Monte Carlo maximum likelihood. In *Computing Science and Statistics, Proceedings of the 23rd Symposium on the Interface, 156.* American Statistical Association.

Ghanem, R. and P. Spanos (2003). *Stochastic finite elements: a spectral approach.* Springer-Verlag.

Girolami, M. and B. Calderhead (2011). Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73*(2), 123–214.

Golightly, A. and D. J. Wilkinson (2011). Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo. *Interface Focus 1*, 807–820.

Gramacy, R. and H. Lee (2012). Cases for the nugget in modeling computer experiments. *Statistics and Computing 22*, 713–722.

Gugushvili, S. and C. A. Klaassen (2012). Root n-consistent parameter estimation for systems of ordinary differential equations: bypassing numerical integration via smoothing. *Bernoulli 18*, 1061–1098.

Hager, W. H. (1989). Updating the inverse of a matrix. *SIAM Review 31*(2), 221–239.

Hennig, P. and S. Hauberg (2014). Probabilistic solutions to differential equations and their application to Riemannian statistics. *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS) (33).*

Henrici, P. (1964). *Elements of Numerical Analysis.* Wiley.

Higham, N. J. (1996). *Accuracy and Stability of Numerical Algorithms* (First ed.). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Huttunen, J. M. J. and J. P. Kaipio (2007). Approximation error analysis in nonlinear state estimation with an application to state-space identification. *Inverse Problems 23*(5), 2141.

Ionides, E., C. Bretó, and A. King (2006). Inference for nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America 103*(49), 18438–18443.

Kaipio, J. and E. Somersalo (2007). Statistical inverse problems: Discretization, model reduction and inverse crimes. *Journal of Computational and Applied Mathematics 198*, 493–504.

Kaipio, J. P., A. Seppanen, E. Somersalo, and H. Haario (2004). Posterior covariance related optimal current patterns in electrical impedance tomography. *Inverse Problems 20*(3), 919.

Keller, H. B. (1968). *Numerical Methods for Two-point Boundary-value Problems.* Blaisdell Publishing Company.

Kennedy, M. (1998). Bayesian quadrature with non-normal approximating functions. *Statistics and Computing 8*, 365 – 375.

Kennedy, M. C. and A. O'Hagan (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society B 63*(3), 425–464.

Leone, F. C., L. S. Nelson, and R. B. Nottingham (1961). The folded normal distribution. *Technometrics 3*, 543–550.

Liang, H. and H. Wu (2008). Parameter estimation for differential equation models using a framework of measurement error in regression models. *Journal of the American Statistical Association 103*(484), 1570–1583.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences 20*, 130–141.

Marjoram, P., J. Molitor, V. Plagnol, and S. Tavaré (2003). Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences 100*(26), 15324–15328.

Marzouk, Y. and H. Najm (2009). Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems. *Journal of Computational Physics 228*(6), 1862–1902.

Marzouk, Y., H. Najm, and L. Rahn (2007). Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics 224*(2), 560–586.

Mosbach, S. and A. G. Turner (2009). A quantitative probabilistic investigation into the accumulation of rounding errors in numerical ODE solution. *Computers & Mathematics with Applications 57*(7), 1157 – 1167.

Neal, R. M. (2011). MCMC using ensembles of states for problems with fast and slow variables such as Gaussian process regression. *arXiv:1101.0387.*

Oberkampf, W. and C. Roy (2010). *Verification and Validation in Scientific Computing.* Cambridge University Press.

O'Hagan, A. O. (1992). Some Bayesian numerical analysis. *Bayesian Statistics 4*, 345–363.

Oliver, T., N. Malaya, R. Ulerich, and R. Moser (2014). Estimating uncertainties in statistics computed from direct numerical simulation. *Physics of Fluids (26).*

Oliver, T. and R. Moser (2011). Bayesian uncertainty quantification applied to RANS turbulence models. *13th European Turbulence Conference (318).*

Osborne, M., D. Duvenaud, R. Garnett, C. Rasmussen, S. J. Roberts, and Z. Ghahramani (2012). Active learning of model evidence using Bayesian quadrature. *Advances in Neural Information Processing Systems 25*, 46–54.

Pellegrini, S. and I. Dusanter-Fourt (1997). The structure, regulation and function of the janus kinases (JAKs) and the signal transducers and activators of transcription (STATs). *European Journal of Biochemistry 248*(3), 615–633.

Polyanin, A. D. and V. F. Zaitsev (2004). *Handbook of Nonlinear Partial Differential Equations.* Chapman and Hall, CRC Press.

Ramsay, J., G. Hooker, and J. Cao (2007). Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society B 69*, 741–796.

Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian Processes for Machine Learning.* Cambridge, Massachusetts: MIT Press.

Raue, A., C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmller, and J. Timmer (2009). Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics 25*, 1923–1929.

Rhode, C. M. (1965). Generalized inverses of partitioned matrices. *Journal of the Society for Industrial and Applied Mathematics 13*(4), 1033–1035.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn (1989). Design and analysis of computer experiments. *Statistical Science 4*(4), 409–423.

Sauer, T., C. Grebogi, and J. A. Yorke (1997). How long do numerical chaotic solutions remain valid? *Physical Review Letters 79*, 59–62.

Schmidl, D., C. Czado, S. Hug, and F. J. Theis (2003). A vine-copula based adaptive MCMC sampler for efficient inference of dynamical systems. *Bayesian Analysis 8*(1), 1–22.

Shampine, L. (2003). Singular boundary value problems for ODEs. *Applied Mathematics and Computation 138*, 99 – 112.

Skilling, J. (1991). *Bayesian Solution of Ordinary Differential Equations*, pp. 23–37. Seattle: Kluwer Academic Publishers.

Stuart, A. M. (2010). Inverse problems: A Bayesian perspective. *Acta Numerica 19*, 451–559.

Swameye, I., T. Muller, J. Timmer, O. Sandra, and U. Klingmuller (2003). Identification of nucleocytoplasmic cycling as a remote sensor in cellular signaling by databased modeling. *Proceedings of the National Academy of Sciences 100*, 1028–1033.

Taylor, S. R. and S. A. Campbell (2007). Approximating chaotic saddles for delay differential equations. *Physical Review E 75*, 046215.

Tokmakian, R., P. Challenor, and Y. Andrianakis (2012). On the use of emulators with extreme and highly nonlinear geophysical simulators. *Journal of Atmospheric and Oceanic Technology 29*, 1704–1715.

Xiu, D. (2009). Fast numerical methods for stochastic computations: a review. *Communications in Computational Physics 5*, 242–272.

Xue, H., H. Miao, and H. Wu (2010). Sieve estimation of constant and time-varying coefficients in nonlinear ordinary differential equation models by considering both numerical error and measurement error. *The Annals of Statistics 38*, 2351–2387.

Xun, X., J. Cao, B. Mallick, R. J. Carroll, and A. Maity (2014). Parameter estimation of partial differential equation models. In press.

8 Appendix

8.1 Problem Specific Algorithms

Algorithm 3 describes a forward simulation strategy for the posterior states of the heat equation (9), where at each temporal grid point (indexed by n) we now need to take into account a discrete spatial component (indexed by m).

Algorithm 5 describes a parallel tempering MCMC algorithm for forward simulation from the two state mixed boundary value problem (2) with boundary constraint $\big(\mathrm{v}(a), \mathrm{u}(b)\big) = \big(\mathrm{v}^*(a), \mathrm{u}^*(b)\big)$. Algorithm 6 describes forward simulation for the delay initial function problem (3), while Algorithm 8 describes a parallel tempering MCMC sampler for drawing realisations from the posterior distribution of the unknowns in the JAK-STAT system problem described in Section 6.4.

We now describe the computational implementation of these probabilistic solvers, as well as Algorithm 1. All of these involve inverting an $n \times n$ covariance matrix at the nth iteration. We suggest instead to use a recursive formulation for updating the sequential matrix inverse (Lemma 8.2), and we have taken this approach in our implementation. This not only provides computational advantages, but it also avoids accumulation of numerical instabilities associated with matrix inversion. Using bounded support covariance structures allows us to further increase efficiency by allowing truncation of the weight matrices employed in the mean and variance updates. Our application to probabilistically solving the Navier-Stokes equations, which comprise over 16,000 coupled ODEs, is an example where taking this approach is vital. In this case, even simply storing matrices of model evaluations quickly exceeds computer memory limits. By employing the truncation technique, we only need to keep track of a fixed number of previous solver iterations, and the updates at each step are exact under a bounded support covariance.

Algorithm 2 uses forward simulation to avoid explicitly computing the joint density of the probabilistic solution and auxiliary model evaluations. Indeed, computation of the forward model is the rate limiting step in the sampling algorithms described for the inverse problem. This is where we can identify some computational savings by exploiting the structure of the probabilistic solver.

Furthermore, it is important to note that our forward simulation algorithms require minimal computational time to draw M additional joint auxiliary and forward model realisations $[\mathrm{u}, \mathbf{f}_{1:N}]$. This fact can be exploited within the inverse problem by using an ensemble MCMC sampling scheme, as proposed by Neal (2011). Indeed, this implementation allows sampling from mixture distributions for which multiple realisations can be quickly computed.

8.2 Choice of Hyperparameters

Within the inverse problem of inference, we may treat the hyperparameters as unknowns to be estimated from the data. We now describe some considerations for the choice of hyperparameters within the forward problem. Hyperparameters may be chosen according to the assumptions used in the consistency result for the probabilistic solver (Theorem 8.4). For this, we suggest setting the prior variance, α^{-1}, and the length-scale, λ, to the order of the maximum step size h over the discretisation grid in each dimension, i.e. let $\alpha^{-1}, \lambda = ch$, where $c > 0$ is a constant related to the shape of the covariance kernel. Under a uniform kernel, a constant of $c = 0.5$ ensures that the resulting step-ahead prediction captures information from exactly one previous model evaluation. The choice $c > 0.5$ provides estimates that are analogous to a multi-step numerical solver with $2c$ steps. Given the peaked shape of the squared exponential kernel, we suggest setting $c > 2$ which places most of the mass of the derivative covariance between the predictive location and the location of the last model evaluation obtained.

Algorithm 3 Sampling from the conditional posterior distribution of u and the $MN \times 1$ stacked vector $\mathbf{f}$ for the heat equation, defined in equation (9), given κ, Ψ, N, M.

```
Define temporal discretisation grid, $\mathbf{s} = [s_1, \cdots, s_N]$, and spatial discretisation grid $\mathbf{z} = [z_1, \cdots, z_M]$.
At time $s_1 := 0$ initialise the second spatial derivative using the given boundary function, $\mathbf{u}_{xx}(\mathbf{z}, 1) := [\mathrm{u}_{xx}(z_1, s_1), \cdots, \mathrm{u}_{xx}(z_M, s_1)]^{\intercal}$. Compute the temporal derivative $\mathbf{f} = \kappa\mathbf{u}_{xx}(\mathbf{z}, 1)$. Define associated model-derivative mismatch $\Lambda := \mathbf{0}_{M \times M}$.
for $n = 1 : N - 1$ do
    Define the predictive mean and variance of the 2nd spatial derivative of the state, using $\otimes$ to denote the Kronecker product for matrices,
        $\mathrm{m}_{xx}(\mathbf{z}, s_{n+1}) = \mathrm{RS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n}) \big(\mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda\big)^{-1} \mathbf{f},$
        $\mathrm{C}_{xx}((\mathbf{z}, s_{n+1}), (\mathbf{z}, s_{n+1})) = \mathrm{RS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{QR}(s_{n+1}, s_{n+1})$
        $-\mathrm{RS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n}) \big(\mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda\big)^{-1} \big(\mathrm{RS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})\big)^{\top}$
    Sample time step-ahead realisation of the 2nd spatial derivative of the state, $\mathbf{u}_{xx}(\mathbf{z}, s_{n+1})$ from the predictive distribution,
        $p\big(\mathbf{u}_{xx}(\mathbf{z}, s_{n+1}) \mid \mathbf{f}, \kappa, \Psi\big) = \mathcal{N}\big(\mathrm{u}_{xx}(\mathbf{z}, s_{n+1}) \mid \mathrm{m}_{xx}(\mathbf{z}, s_{n+1}), \mathrm{C}_{xx}((\mathbf{z}, s_{n+1}), (\mathbf{z}, s_{n+1}))\big);$
    Augment vector $\mathbf{f} := [\mathbf{f}, \kappa\mathbf{u}_{xx}(\mathbf{z}, s_{n+1})]$, and augment model-derivative mismatch matrix $\Lambda := \mathrm{diag}\big\{\mathrm{diag}\{\Lambda\}, C_t((\mathbf{z}, s_{n+1}), (\mathbf{z}, s_{n+1}))\big\}$ with predictive derivative variance, where
        $\mathrm{C}_t((\mathbf{z}, s_{n+1}), (\mathbf{z}, s_{n+1})) = \mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(s_{n+1}, s_{n+1})$
        $- \mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(s_{n+1}, \mathbf{s}_{1:n}) \big(\mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda\big)^{-1} \mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:n}, s_{n+1})$
end for
Define,
        $\mathrm{m}(\cdot, \cdot) = \mathrm{SS}(\cdot, \mathbf{z}) \otimes \mathrm{QR}(\cdot, \mathbf{s}_{1:N}) \big(\mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda\big)^{-1} \mathbf{f},$
        $\mathrm{C}((\cdot, \cdot), (\cdot, \cdot)) = \mathrm{SS}(\cdot, \cdot) \otimes \mathrm{QR}(\cdot, \cdot)$
        $- \mathrm{SS}(\cdot, \mathbf{z}) \otimes \mathrm{QR}(\cdot, \mathbf{s}_{1:N}) \big(\mathrm{SS}(\mathbf{z}, \mathbf{z}) \otimes \mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda\big)^{-1} \big(\mathrm{SS}(\cdot, \mathbf{z}) \otimes \mathrm{QR}(\cdot, \mathbf{s}_{1:N})\big)^{\top}$
Return both $\mathrm{u} \sim \mathcal{GP}\big(\mathrm{m}, \mathrm{C}\big)$ and $\mathbf{f}$.
```

Algorithm 4 Metropolis-Hastings algorithm to draw K samples from the N-dimensional probabilistic MBVP solution $p\big(u(\mathbf{t}), v(\mathbf{t}) \mid \mathrm{u}^*(b), v^*(a), \theta, \Psi, N\big)$

```
Initialise the unknown boundary value $\mathrm{u}(a)$ and conditionally simulate probabilistic solution realisation $\big(\mathrm{u}(\mathbf{t}), v(\mathbf{t}), \mathbf{f}_{1:N}\big) \sim p\big(\cdot \mid v^*(a), u(a), \theta, \Psi, N\big)$ via Algorithm 1.
for $k = 1 : K$ do
  Propose boundary value $\mathrm{u}'(a) \sim q(\cdot \mid \mathrm{u}(a))$ where $q$ is a proposal density.
  Use Algorithm 1 to conditionally simulate probabilistic solution,
      $\big(\mathrm{u}'(\mathbf{t}), v'(\mathbf{t}), \mathbf{f}_{1:N}\big) \sim p\big(\cdot \mid v^*(a), u'(a), \theta, \Psi, N\big)$
  Compute,
      $\rho = \frac{q\big(\mathrm{u}'(a) \mid \mathrm{u}(a)\big)}{q\big(\mathrm{u}(a) \mid \mathrm{u}(a)\big)} \frac{p(\mathrm{u}'(a))}{p(\mathrm{u}(a))} \left\{ \frac{p\big(\mathrm{u}'(b) \mid \mathrm{u}^*(b)\big)}{p\big(\mathrm{u}(b) \mid \mathrm{u}^*(b)\big)} \right\}$
  if $\min\{1, \rho\} > \mathrm{U}[0,1]$ then
    Update $\mathrm{u}(a) = \mathrm{u}'(a)$.
    Update $\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t})\big) = \big(\mathrm{u}'(\mathbf{t}), v'(\mathbf{t})\big)$.
  end if
  Return $\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t})\big)$.
end for
```

We note that the choice of hyperparameters within the forward problem is not a new question. Indeed, existing numerical methods, by choosing the step number or the type of solver, effectively fix the analogues of these hyperparameters, and indeed maintain them fixed within the inverse problem. In the JAK-STAT example we note, however, that doing this may introduce bias when the solution approximation suffers from low numerical accuracy. In contrast, our framework not only allows us to estimate optimal choices of the solver by estimating hyperparameters instead of holding them fixed, but also allows us to consider cases where the optimal solver settings vary across the parameter space.

8.3 Some Kernels and their Convolutions

Choice of the covariance function is related to the assumed smoothness of the exact but unknown solution. The examples in this paper utilise two types of covariance functions, although the results are more generally applicable. The squared exponential covariance, obtained by convolving the kernel,

$$R_\lambda(t_1, t_2) = \exp\left\{ -\frac{(t_1 - t_2)^2}{2\lambda^2} \right\},$$

is infinitely differentiable. Thus, we utilise this covariance structure in the solution of the Lorenz system, the Navier-Stokes equations, the Lane-Emden mixed boundary value problem, and the heat equation. In contrast, solutions for systems of delay initial function problems are often characterised by second derivative discontinuities at the lag locations. In order to avoid bias from over-smoothing, we utilise the uniform covariance structure for modelling the derivative. This covariance, obtained by convolving the kernel,

$$R_\lambda(t_1, t_2) = \mathbb{I}\{t_2 \in (t_1 - \lambda, t_1 + \lambda)\},$$

is non-differentiable.

Closed form expressions for the pairwise convolutions for the two covariance functions are provided below. Let R_λ be the squared exponential kernel and let Q_λ be its integrated version. Then,

$$\alpha \mathrm{RR}(t_1, t_2) \quad = \quad \sqrt{\pi}\lambda \exp\left\{ -\frac{(t_1 - t_2)^2}{4\lambda^2} \right\}$$

Algorithm 5 Parallel tempering algorithm with C chains to draw K samples from the N-dimensional probabilistic MBVP solution, i.e. from the distribution $p\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t}) \mid \mathrm{u}^*(b), \mathrm{v}^*(a), \theta, \Psi, N\big)$

```
Initialise unknown boundary values, $\mathrm{u}(a)_{(c)}$, for each chain, $c = 1, \ldots, C$.
Define the probability $s$ of performing a swap move between 2 randomly chosen chains at each iteration,
and define a temperature vector $\gamma \in (0,1]^C$, such that $\gamma_C = 1$.
Use Algorithm 4 to conditionally simulate,
    $\big(\mathrm{u}(\mathbf{t}), \mathrm{v}(\mathbf{t})\big) \sim p\big(\cdot \mid \mathrm{u}^*(b), v^*(a), \theta, \Psi, N\big)$

for $k = 1 : K$ do
  if $s > \mathrm{U}[0,1]$ then
    Propose a swap between $i, j \sim q(i,j)$, such that $i \neq j$.
    Compute:
      $\rho = \frac{\{p(\mathrm{u}(b)_{(i)} \mid \mathrm{u}^*(b))\}^{\gamma_j}}{\{p(\mathrm{u}(b)_{(i)} \mid \mathrm{u}^*(b))\}^{\gamma_i}} \frac{\{p(\mathrm{u}(b)_{(j)} \mid \mathrm{u}^*(b))\}^{\gamma_i}}{\{p(\mathrm{u}(b)_{(j)} \mid \mathrm{u}^*(b))\}^{\gamma_j}}$

    if $\min(1, \rho) > \mathrm{U}[0,1]$ then
      Swap initial conditions $\mathrm{u}(a)_{(i)}$ and $\mathrm{u}(a)_{(j)}$.
    end if
  end if

  for $c = 1 : C$ do
    Perform one iteration of Metropolis-Hastings Algorithm 4, using a tempered likelihood with temperature $\gamma_c$ and initial condition $\mathrm{u}(a)_{(c)}$.
  end for

  Return $\big(\mathrm{u}(\mathbf{t})_{(C)}, \mathrm{v}(\mathbf{t})_{(C)}\big)$.
end for
```

$$\begin{aligned}
\alpha\mathrm{QR}(t_1,t_2) &= \pi\lambda^2 \mathrm{erf}\left\{\frac{t_1-t_2}{2\lambda}\right\} + \pi\lambda^2 \mathrm{erf}\left\{\frac{t_2-a}{2\lambda}\right\} \\
\alpha\mathrm{QQ}(t_1,t_2) &= \pi\lambda^2(t_1-a)\mathrm{erf}\left\{\frac{t_1-a}{2\lambda}\right\} + 2\sqrt{\pi}\lambda^3 \exp\left\{-\frac{(t_1-a)^2}{4\lambda^2}\right\} \\
&\quad -\pi\lambda^2(t_2-t_1)\mathrm{erf}\left\{\frac{t_2-t_1}{2\lambda}\right\} - 2\sqrt{\pi}\lambda^3 \exp\left\{-\frac{(t_2-t_1)^2}{4\lambda^2}\right\} \\
&\quad +\pi\lambda^2(t_2-a)\mathrm{erf}\left\{\frac{t_2-a}{2\lambda}\right\} + 2\sqrt{\pi}\lambda^3 \exp\left\{-\frac{(t_2-a)^2}{4\lambda^2}\right\} - 2\sqrt{\pi}\lambda^3
\end{aligned}$$

Next, let R_λ be the uniform kernel and let Q_λ be its integrated version. Then,

$$\begin{aligned}
\alpha\mathrm{RR}(t_1,t_2) &= \{\min(t_1,t_2) - \max(t_1,t_2) + 2\lambda\}\, \mathrm{I}\{\min(t_1,t_2) - \max(t_1,t_2) > -2\lambda\} \\
\alpha\mathrm{QR}(t_1,t_2) &= 2\lambda x\Big|_{\max(a+\lambda, t_2-\lambda)}^{\min(t_1-\lambda, t_2+\lambda)} + \left\{\frac{x^2}{2} + (\lambda - a)x\right\}\Bigg|_{t_2-\lambda}^{\min(a+\lambda, t_1-\lambda, t_2+\lambda)} \\
&\quad + \left\{(t_1+\lambda)x - \frac{x^2}{2}\right\}\Bigg|_{\max(a+\lambda, t_1-\lambda, t_2-\lambda)}^{\min(t_1,t_2)+\lambda} \\
&\quad +[(t_1-a)\{\max(t_1,t_2) - a - 2\lambda\}]\mathrm{I}\{a - \max(t_1,t_2) > -2\lambda\}, \\
\alpha\mathrm{QQ}(t_1,t_2) &= 4\lambda^2\{\min(t_1,t_2) - a - 2\lambda\}\mathrm{I}\{\min(t_1,t_2) > a + 2\lambda\} + 2\lambda\left\{(t_2+\lambda)x - \frac{x^2}{2}\right\}\Bigg|_{\max(a+\lambda, t_2-\lambda)}^{\min(t_1-\lambda, t_2+\lambda)}
\end{aligned}$$

Algorithm 6 Sampling from the joint posterior distribution of u and $\mathbf{f}_{1:N}$ for an ODE delay initial function problem, defined by equation (3), given $\phi, \tau, \theta, \Psi, N$.

```
At time $s_1 := a$, initialise the derivative $\mathrm{f}_1 := f\big(s_1, \mathrm{u}(s_1), \phi(s_1 - \tau), \theta\big)$ for initial state $\mathrm{u}(s_1) := \phi(a)$, and
define associated model-derivative mismatch matrix, $\Lambda_{1\times 1} := 0$.
for $n = 1 : N - 1$ do
    Define the predictive state mean and variance,

        $\mathrm{m}(s_{n+1}) = \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathbf{f}_{1:n},$
        $\mathrm{C}(s_{n+1}, s_{n+1}) = \mathrm{QQ}(s_{n+1}, s_{n+1}) - \mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{QR}(s_{n+1}, \mathbf{s}_{1:n})^\top$

    Sample step-ahead realisation $\mathrm{u}(s_{n+1})$ from the predictive distribution of the state,

        $p\big(\mathrm{u}(s_{n+1}) \mid \mathbf{f}_{1:n}, \phi, \tau, \theta, \Psi\big) = \mathcal{N}\big(\mathrm{u}(s_{n+1}) \mid \mathrm{m}(s_{n+1}), \mathrm{C}(s_{n+1}, s_{n+1})\big)$

    if $s_{n+1} - \tau \geq a$ then
        Compute the lagged mean $\mathrm{u}(s_{n+1} - \tau) = \mathrm{m}(s_{n+1} - \tau)$.
    else
        Compute the lagged mean $\mathrm{u}(s_{n+1} - \tau) = \phi(s_{n+1} - \tau)$.
    end if

    Evaluate the ODE model $\mathrm{f}_{n+1} := f(s_{n+1}, \mathrm{u}(s_{n+1}), \mathrm{u}(s_{n+1} - \tau), \theta)$ for realisations $\mathrm{u}(s_{n+1})$ and $\mathrm{u}(s_{n+1} - \tau)$
    at the next time point, $s_{n+1}$, and augment the vector $\mathbf{f}_{1:n} := [\mathbf{f}_{1:n}, \mathrm{f}_{n+1}]$.
    Define the predictive derivative covariances,

        $C_t(s_{n+1}, s_{n+1}) = \mathrm{RR}(s_{n+1}, s_{n+1}) - \mathrm{RR}(s_{n+1}, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{RR}(s_{n+1}, \mathbf{s}_{1:n}),$
        $C_t(s_{n+1} - \tau, s_{n+1} - \tau) = \mathrm{RR}(s_{n+1} - \tau, s_{n+1} - \tau) - \mathrm{RR}(s_{n+1} - \tau, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{RR}(\mathbf{s}_{1:n}, s_n - \tau),$
        $C_t(s_{n+1} - \tau, s_{n+1}) = \mathrm{RR}(s_{n+1} - \tau, s_{n+1}) - \mathrm{RR}(s_{n+1} - \tau, \mathbf{s}_{1:n})\big(\mathrm{RR}(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}) + \Lambda_{n\times n}\big)^{-1}\mathrm{RR}(\mathbf{s}_{1:n}, s_n)$

    and augment the matrix, $\Lambda_{(n+1)\times(n+1)} := \mathrm{diag}\big\{\mathrm{diag}\{\Lambda_{n\times n}\}, \mathrm{C}_t(s_{n+1}, s_{n+1}) + \mathrm{C}_t(s_{n+1} - \tau, s_{n+1} - \tau) +$
    $2\mathrm{C}_t(s_{n+1}, s_{n+1} - \tau)\big\}$.
end for

Define,

        $m(\cdot) = \mathrm{QR}(\cdot, \mathbf{s}_{1:N})\big(\mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda_{N\times N}\big)^{-1}\mathbf{f}_{1:N},$
        $C(\cdot, \cdot) = \mathrm{QQ}(\cdot, \cdot) - \mathrm{QR}(\cdot, \mathbf{s}_{1:N})\big(\mathrm{RR}(\mathbf{s}_{1:N}, \mathbf{s}_{1:N}) + \Lambda_{N\times N}\big)^{-1}\mathrm{QR}(\cdot, \mathbf{s}_{1:N})^\top$

Return both $\mathrm{u} \sim \mathcal{GP}(\mathrm{m}, \mathrm{C})$ and $\mathbf{f}_{1:N}$.
```

Algorithm 7 Metropolis-Hastings algorithm to draw K samples from the posterior distribution with density $p\big(\theta, \tau, \phi^{(1)}(0), \mathrm{u}(\mathbf{t};\theta), \alpha, \lambda \mid y(\mathbf{t}), \Sigma, \Psi, N\big)$ for an ODE delay initial function problem.

```
Initialise $\theta, \tau, \phi^{(1)}(0), \alpha, \lambda$.
for $k = 1 : K$ do
  Propose $\theta', \tau', \phi^{(1)'}(0) \sim q(\cdot \mid \theta, \tau, \phi^{(1)}(0))$ where $q$ is a proposal density.
  Use Algorithm 6 to conditionally simulate probabilistic solution,
      $\big(\mathrm{u}'(\mathbf{t}, \theta'), \mathbf{f}_{1:N}\big) \sim p\big(\cdot \mid \theta', \tau', \phi^{(1)'}(0), \alpha, \lambda, \Psi, N\big)$
  Compute,
      $\rho = \dfrac{q(\,\theta'\,, \tau', \phi^{(1)'}(0) \mid \theta, \tau, \phi^{(1)}(0)\,)}{q(\,\theta\,, \tau, \phi^{(1)}(0) \mid \theta', \tau', \phi^{(1)'}(0)\,)} \, \dfrac{p(\,\theta'\,, \tau', \phi^{(1)}(0)'\,)}{p(\,\theta, \tau, \phi^{(1)}(0)\,)} \left\{ \dfrac{p\big(\,y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}'(\mathbf{t}, \theta'), \theta'\big), \Sigma\,\big)}{p\big(\,y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}(\mathbf{t}, \theta), \theta\big), \Sigma\,\big)} \right\}$
  if $\min(1, \rho) > \mathrm{U}[0, 1]$ then
    Update $\big(\theta, \tau, \phi^{(1)}(0)\big) = \big(\theta'\,, \tau', \phi^{(1)'}(0)\big)$.
  end if
  Propose $\alpha', \lambda' \sim q(\cdot \mid \alpha, \lambda)$.
  Use Algorithm 6 to conditionally simulate probabilistic solution,
      $\big(\mathrm{u}'(\mathbf{t}, \theta), \mathbf{f}_{1:N}\big) \sim p\big(\cdot \mid \theta, \tau, \phi^{(1)}(0), \alpha', \lambda', \Psi, N\big)$
  Compute,
      $\rho = \dfrac{q(\alpha', \lambda' \mid \alpha, \lambda)}{q(\alpha, \lambda \mid \alpha', \lambda')} \dfrac{p(\alpha', \lambda')}{p(\alpha, \lambda)} \left\{ \dfrac{p\big(\,y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}'(\mathbf{t}, \theta), \theta\big), \Sigma\,\big)}{p\big(\,y(\mathbf{t}) \mid \mathcal{G}\big(\mathrm{u}(\mathbf{t}, \theta), \theta\big), \Sigma\,\big)} \right\}$
  if $\min(1, \rho) > \mathrm{U}[0, 1]$ then
    Update $\big(\alpha, \lambda\big) = \big(\alpha', \lambda'\big)$;
  end if
  Return $\big(\theta, \tau, \phi^{(1)}(0), \mathrm{u}(\mathbf{t}, \theta), \alpha, \lambda\big)$.
end for
```

$$
\begin{aligned}
&+\left\{\frac{x^3}{3} + (\lambda - a)x^2 + (\lambda - a)^2 x\right\}\Bigg|_{a-\lambda}^{\min(a+\lambda, t_1-\lambda, t_2-\lambda)} \\
&+(t_2 - a)\left\{\frac{x^2}{2} + (\lambda - a)x\right\}\Bigg|_{t_2-\lambda}^{\min(a+\lambda, t_1-\lambda)} + 2\lambda\left\{(t_1 + \lambda)x - \frac{x^2}{2}\right\}\Bigg|_{\max(a+\lambda, t_1-\lambda)}^{\min(t_1+\lambda, t_2-\lambda)} \\
&+\left\{(t_1 + \lambda)(t_2 + \lambda)x - (t_1 + t_2 + 2\lambda)\frac{x^2}{2} + \frac{x^3}{3}\right\}\Bigg|_{\max(a+\lambda, t_1-\lambda, t_2-\lambda)}^{\min(t_1, t_2)+\lambda} \\
&+(t_1 - a)\left\{\frac{x^2}{2} + (\lambda - a)x\right\}\Bigg|_{t_1-\lambda}^{\min(a+\lambda, t_2-\lambda)} \\
&+(t_1 - a)(t_2 - a)\{a + 2\lambda - \max(t_1, t_2)\}\mathrm{I}\{a + 2\lambda > \max(t_1, t_2)\}.
\end{aligned}
$$

8.4 Probabilistic Solution as Latent Function Estimation

We take a latent function view of the model presented in Section 4. Let $\mathcal{F} = L^2\left(\mathbb{R}; (H, \mathcal{A}, \mu_0)\right)$ be the space of square-integrable random functions and $\mathcal{F}^*$ be its dual space of linear functionals. The solution

Algorithm 8 Parallel tempering algorithm with C chains to draw K samples from the posterior distribution with density $p\big(\theta, \tau, \phi^{(1)}(0), \mathrm{u}(\mathbf{t}, \theta), \alpha, \lambda \mid y(\mathbf{t}), \Sigma, \Psi, N\big)$ for an ODE delay initial function problem.

```
Initialise parameters $\theta_{(c)}, \tau_{(c)}, \phi^{(1)}_{(c)}(0), \alpha_{(c)}, \lambda_{(c)}$, for each chain, $c = 1, \dots, C$.
Define the probability $s$ of performing a swap move between 2 randomly chosen chains at each iteration,
and define a temperature vector $\gamma \in (0,1]^C$, such that $\gamma_C = 1$.
Use Algorithm 6 to conditionally simulate,
    $\big(u_{(c)}(\mathbf{t}, \theta_{(c)}), \mathbf{f}_{1:N,(c)}\big) \sim p\big(\cdot \mid \theta_{(c)}, \tau_{(c)}, \phi^{(1)}_{(c)}(0), \alpha_{(c)}, \lambda_{(c)}, \Psi, N\big)$

for $k = 1 : K$ do
  if $s > \mathrm{U}[0,1]$ then
    Propose a swap between $i, j \sim q(i,j)$, such that $i \neq j$;
    Compute:
      $\rho = \dfrac{\big\{p\big(\, y(\mathbf{t}) \mid \mathcal{G}\big(u_{(i)}(\mathbf{t}, \theta_{(i)}), \theta_{(i)}\big), \Sigma\,\big)\big\}^{\gamma_j} \; \big\{p\big(\, y(\mathbf{t}) \mid \mathcal{G}\big(u_{(j)}(\mathbf{t}, \theta_{(j)}), \theta_{(j)}\big), \Sigma\,\big)\big\}^{\gamma_i}}{\big\{p\big(\, y(\mathbf{t}) \mid \mathcal{G}\big(u_{(i)}(\mathbf{t}, \theta_{(i)}), \theta_{(i)}\big), \Sigma\,\big)\big\}^{\gamma_i} \; \big\{p\big(\, y(\mathbf{t}) \mid \mathcal{G}\big(u_{(j)}(\mathbf{t}, \theta_{(j)}), \theta_{(j)}\big), \Sigma\,\big)\big\}^{\gamma_j}};$

    if $\min(1, \rho) > \mathrm{U}[0,1]$ then
      Swap parameters $\big(\theta_{(i)}, \tau_{(i)}, \phi^{(1)}_{(i)}(0), \alpha_{(i)}, \lambda_{(i)}\big)$ and $\big(\theta_{(j)}, \tau_{(j)}, \phi^{(1)}_{(j)}(0), \alpha_{(j)}, \lambda_{(j)}\big)$;
    end if
  end if

  for $c = 1 : C$ do
    Perform one iteration of Metropolis-Hastings Algorithm 7, using a tempered likelihood with temper-
    ature $\gamma_c$ and initialised parameters $\theta_{(c)}, \tau_{(c)}, \phi^{(1)}_{(c)}(0), \Psi_{(c)}$.
  end for
  Return $\big(\theta_{(C)}, \tau_{(C)}, \phi^{(1)}_{(C)}(0), \Psi_{(C)}, u_{(C)}(\mathbf{t}, \theta_{(C)})\big)$.
end for
```

and its derivative will be modelled by an integral transform using the linear continuous operators R and Q defining a mapping from $\mathcal{F}$ to $\mathcal{F}^*$. The associated kernels are the deterministic, square-integrable function $R_\lambda : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ and its integrated version $Q_\lambda(t_1, t_2) := \int_a^{t_2} R_\lambda(s, t_2)\mathrm{d}s$. The operators $\mathrm{R}, \mathrm{Q}, \mathrm{R}^\dagger, \mathrm{Q}^\dagger$ are defined, for $\mathrm{u} \in \mathcal{F}$ and $\mathrm{v} \in \mathcal{F}^*$, as $\mathrm{Ru}(t) = \int R_\lambda(t,z)\mathrm{u}(z)\mathrm{d}z$ and $\mathrm{Qu}(t) = \int Q_\lambda(t,z)\mathrm{u}(z)\mathrm{d}z$, with adjoints $\mathrm{R}^\dagger\mathrm{v}(t) = \int R_\lambda(z,t)\mathrm{v}(z)\mathrm{d}z$ and $\mathrm{Q}^\dagger\mathrm{v}(t) = \int Q_\lambda(z,t)\mathrm{v}(z)\mathrm{d}z$ respectively.

As suggested in Skilling (1991), solving a differential equation problem may be restated as a problem of estimating an underlying latent process $\zeta \in \mathcal{F}$. We consider the white noise process, $\zeta \sim \mathcal{N}(0, K)$, with covariance $K(t_1, t_2) = \alpha^{-1}\delta_{t_1}(t_2)$. Next, we model the derivative of the solution as the integral transform,

$$\mathrm{u}_t(t) := \mathrm{m}^0_t(t) + \mathrm{R}\zeta(t), \quad t \in [a,b]. \tag{18}$$

The differential equation solution model is then obtained by integrating $\mathrm{u}_t(t)$ with respect to t,

$$\mathrm{u}(t) = \int_a^t \mathrm{u}_s(s)\mathrm{d}s = \mathrm{m}^0(t) + \mathrm{Q}\zeta(t), \quad t \in [a,b]. \tag{19}$$

Our goal is to update the prior $\mathrm{u}^0(t)$ given a vector of N noisy derivative realisations, $\mathbf{f}_{1:N}$, to obtain the probabilistic solution, which we denote $\mathrm{u}^N(t)$.

Lemma 8.1 (Smoothing)**.** *The probabilistic solution $\{u^N(t), t \in [a,b]\}$ and its time derivative $\{u^N_t(t), t \in [a,b]\}$ are well-defined and distributed according to Gaussian measures μ^N and μ^N_t with mean functions m^N, m^N_t and covariance operators C^N, C^N_t respectively, given by,*

$$m^N(t_1) = m^0(t_1) + \int_a^{t_1} C^0_t(s, \mathbf{s}_{1:N})\, ds\, \big(\Lambda_{N\times N} + C^0_t(\mathbf{s}_{1:N}, \mathbf{s}_{1:N})\big)^{-1} \big(\mathbf{f}_{1:N} - m^0_t(\mathbf{s}_{1:N})\big),$$

$$
\begin{aligned}
m_t^N(t_1) &= m_t^0(t_1) + C_t^0(t_1, \mathbf{s}_{1:N})\ \big(\Lambda_{N\times N} + C_t^0(\mathbf{s}_{1:N}, \mathbf{s}_{1:N})\big)^{-1}\ \big(\mathbf{f}_{1:N} - m_t^0(\mathbf{s}_{1:N})\big),\\
C^N(t_1,t_2) &= C^0(t_1,t_2) + \int_a^{t_1} C_t^0(s, \mathbf{s}_{1:N})\, ds\, \big(\Lambda_{N\times N} + C_t^0(\mathbf{s}_{1:N}, \mathbf{s}_{1:N})\big)^{-1} \int_a^{t_2} C_t^0(\mathbf{s}_{1:N}, s)\, ds,\\
C_t^N(t_1,t_2) &= C_t^0(t_1,t_2) + C_t^0(t_1, \mathbf{s}_{1:N})\ \big(\Lambda_{N\times N} + C_t^0(\mathbf{s}_{1:N}, \mathbf{s}_{1:N})\big)^{-1}\ C_t^0(\mathbf{s}_{1:N}, t_2),
\end{aligned}
$$

where m^0 and m_t^0 are the prior means and C^0 and C_t^0 the prior covariances of the state and derivatives.

Proof. We are interested in the conditional distribution of the state $\mathrm{u}(t) - \mathrm{m}^0(t) \in \mathcal{F}^*$ and time derivative $\mathrm{u}_t(t) - \mathrm{m}_t^0(t) \in \mathcal{F}^*$ given a vector of N noisy derivative evaluations on a mesh with vertices $\mathbf{s}_{1:N} \in [a,b]^N$ under the Gaussian error model,

$$\mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N}) = \mathrm{R}\zeta(\mathbf{s}_{1:N}) + \eta(\mathbf{s}_{1:N}),$$

where $\eta(\mathbf{s}_{1:N}) \sim \mathcal{N}_n(\mathbf{0}, \Lambda_{N\times N})$ is independent of ζ, and $\Lambda_{N\times N} \in \mathrm{M}^N(\mathbb{R})$ is a positive definite matrix.

Construct the vector

$$[\mathrm{u}_t - \mathrm{m}_t^0, \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})] = [\mathrm{R}\zeta, \mathrm{R}\zeta(\mathbf{s}_{1:N}) + \eta(\mathbf{s}_{1:N})] \in \mathcal{F}^* \oplus \mathbb{R}^N,$$

where the first element is function-valued and the second element is vector-valued. This vector is jointly Gaussian with mean $M = (0, \mathbf{0})$ and covariance operator C with positive definite cross-covariance operators,

$$
\begin{aligned}
C_{11} &= \mathrm{R}K\mathrm{R}^\dagger & C_{12} &= \mathrm{R}K\mathrm{R}^\dagger\\
C_{21} &= \mathrm{R}K\mathrm{R}^\dagger & C_{22} &= \mathrm{R}K\mathrm{R}^\dagger + \Lambda_{n\times n}.
\end{aligned}
\tag{20}
$$

Since both $\mathcal{F}^*$ and $\mathbb{R}^N$ are separable Hilbert spaces, it follows from Theorem 6.20 in Stuart (2010) that the random variable $[\mathrm{u}_t - \mathrm{m}_t^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})]$ is well-defined and distributed according to a Gaussian measure with mean and covariance,

$$
\begin{aligned}
\mathrm{E}\big[\mathrm{u}_t - \mathrm{m}_t^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})\big] &= C_{12}C_{22}^{-1}(\mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})),\\
\mathrm{Cov}\big[\mathrm{u}_t - \mathrm{m}_t^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})\big] &= C_{11} - C_{12}C_{22}^{-1}C_{21}.
\end{aligned}
$$

Similarly, we consider the vector $[u - \mathrm{m}^0, \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})] = [\mathrm{Q}\zeta, \mathrm{R}\zeta(\mathbf{s}_{1:N}) + \eta(\mathbf{s}_{1:N})] \in \mathcal{F}^* \oplus \mathbb{R}^n$, with mean $M = (0, \mathbf{0})$ and cross-covariances,

$$
\begin{aligned}
C_{11} &= \mathrm{Q}K\mathrm{Q}^\dagger & C_{12} &= \mathrm{Q}K\mathrm{R}^\dagger\\
C_{21} &= \mathrm{R}K\mathrm{Q}^\dagger & C_{22} &= \mathrm{R}K\mathrm{R}^\dagger + \Lambda_{n\times n}.
\end{aligned}
$$

By Theorem 6.20 (Stuart, 2010), the conditional distribution of $[\mathrm{u}_t - \mathrm{m}_t^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})]$ is a well-defined Gaussian distribution with mean and covariance,

$$
\begin{aligned}
\mathrm{E}\big[\mathrm{u} - \mathrm{m}_t^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})\big] &= C_{12}C_{22}^{-1}(\mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})),\\
\mathrm{Cov}\big[\mathrm{u} - \mathrm{m}^0 \mid \mathbf{f}_{1:N} - \mathrm{m}_t^0(\mathbf{s}_{1:N})\big] &= C_{11} - C_{12}C_{22}^{-1}C_{21}.
\end{aligned}
$$

□

A necessary condition for a well-defined probabilistic solution derivative is that the cross-covariance operators, (20), between the derivative and n derivative model realisations, be positive definite. We note that this condition is trivially satisfied by kernels R_λ that are not everywhere zero.

8.5 Probabilistic Solution as Sequential Bayesian Updating

In this section we exploit a result originally presented in Hager (1989) to avoid computationally expensive matrix inversion required to obtain the probabilistic IVP solution (equation 1) using Algorithm 1. We then present Lemma 8.3, which uses this idea to formulate Algorithm 1 as a sequential Bayesian updating procedure, which will subsequently allow us to show consistency of the probabilistic solution in Section 8.6.

Lemma 8.2 (Fast Covariance Inversion)**.** *For a stationary kernel R_λ and non-negative diagonal $n \times n$ matrix $\Lambda_{n\times n}$, the matrix inverse $(B^n)^{-1} := (\Lambda_{n\times n} + RR(\mathbf{s}_{1:n}, \mathbf{s}_{1:n}))^{-1}$ can be written recursively in terms of $(B^{n-1})^{-1}$ as*

$$\begin{aligned}
(B^n)^{-1}_{(1:n-1,1:n-1)} &= (B^{n-1})^{-1} \, \frac{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n) + RR(\mathbf{s}_{1:n-1}, s_n) RR(s_n, \mathbf{s}_{1:n-1}) (B^{n-1})^{-1}}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)}, \\
(B^n)^{-1}_{(n,1:n-1)} &= -(B^{n-1})^{-1} \, \frac{RR(\mathbf{s}_{1:n-1}, s_n)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)}, \\
(B^n)^{-1}_{(1:n-1,n)} &= -(B^{n-1})^{-1} \, \frac{RR(\mathbf{s}_{1:n-1}, s_n)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)}, \\
(B^n)^{-1}_{(n,n)} &= \frac{1}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)}.
\end{aligned}$$

Proof. Since B^n is a non-negative symmetric partitioned matrix, we can write its inverse in block form (see for example, Hager, 1989; Rhode, 1965). □

Lemma 8.3 (Prediction)**.** *The probabilistic IVP solution and its derivative at the n'th ($1 \leq n \leq N$) iteration of Algorithm 1 are Gaussian with mean and covariance that can be expressed recursively as,*

$$\begin{aligned}
m^n(t_1) &= m^{n-1}(t_1) + (f_n - m_t^{n-1}(s_n)) \int_a^{t_1} \frac{C_t^{n-1}(s, s_n)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)} ds, \\
m_t^n(t_1) &= m_t^{n-1}(t_1) + (f_n - m_t^{n-1}(s_n)) \, \frac{C_t^{n-1}(t_1, s_n)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)}, \\
C^n(t_1, t_2) &= C^{n-1}(t_1, t_2) - \int_a^{t_1} \int_a^{t_2} \frac{C_t^{n-1}(z, s_n) \, C_t^{n-1}(s_n, s)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)} ds dz, \\
C_t^n(t_1, t_2) &= C_t^{n-1}(t_1, t_2) - \frac{C_t^{n-1}(t_1, s_n) \, C_t^{n-1}(s_n, t_2)}{\Lambda^{(n,n)}_{n\times n} + C_t^{n-1}(s_n, s_n)},
\end{aligned}$$

where m^0 and m_t^0 are the prior means and C^0 and C_t^0 the prior covariances of the state and derivatives.

Proof. The proof is similar to Lemma 8.1, where we take the prior for the nth update to be the posterior measure obtained at the previous iteration of Algorithm 1. The conditioning is now made on the single nth evaluation $f_n := f\big(s_n, u^{n-1}(s_n), \theta\big)$. That is, we take $m^0 := m^{n-1}$, $m_t^0 := m_t^{n-1}$, $\mathrm{C}^0 := \mathrm{C}^{n-1}$, and $\mathrm{C}_t^0 := \mathrm{C}_t^{n-1}$. An informal derivation can be obtained by using the expressions for the means and covariances in Lemma 8.1 together with the recursive matrix inverse obtained in Lemma 8.2. □

8.6 Consistency of the probabilistic solution

Theorem 8.4 (Consistency)**.** *Consider initial value problem (1) on $[a, b]$ with $f(\cdot, \cdot, \theta) : [a, b] \times \mathbb{R} \to \mathbb{R}$ continuous in the first argument and Lipschitz continuous in the second argument. Further assume that its exact solution, belonging to a Hilbert space H, is continuously differentiable and has bounded second order remainders on the Taylor series expansion. The stochastic process obtained using Algorithm 1 under a covariance kernel satisfying conditions (23) and (24), converges in L^1 to the unique solution satisfying (1) if $\mu^0(H) = 1$.*

Proof. Denote the exact solution satisfying (1) by $\mathrm{u}^*(t)$. For clarity of exposition, we assume that $\mathrm{u}(a) = 0$ and let $\mathrm{m}^0(t) = 0$ for all $t \in [a, b]$. We define $h = \max_{n=2,\ldots,N} \big(s_n - s_{n-1}\big)$ to be the maximum step length between nearby discretisation grid points. We would like to show that the probabilistic solution $\{\mathrm{u}^N(t), t \in [a, b]\}$ is a consistent estimator of the exact solution $\mathrm{u}^*(t)$ when the vector $\mathbf{f}_{1:N}$ is built up sequentially using Algorithm 1 as $h \to 0$ and $\lambda, \alpha^{-1} = O(h)$. Solution updates are constructed using the recurrence derived in Lemma 8.3, under the limiting assumption (Lemma 8.7) that $\Lambda = \mathbf{0}$.

For a given $t \in [a,b]$ we find n such that $t \in [s_n, s_{n+1}]$, and bound the expected absolute difference between the nth probabilistic solution and the exact solution as follows,

$$\begin{aligned}
\beta_n(t) &:= \mathrm{E}\big\{|\mathrm{u}(t) - \mathrm{u}^*(t)| \;\big|\; \mathbf{f}_{1:n}, \theta, \Psi\big\} \\
&= \mathrm{E}\big[\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big] \left\{1 - 2\Phi\left(-\frac{\mathrm{E}\big[\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big]}{\sqrt{\mathrm{C}^n(t,t)}}\right)\right\} \\
&\qquad + \sqrt{\frac{2}{\pi}\,\mathrm{C}^n(t,t)}\ \exp\left\{-\frac{\big(\mathrm{E}\big[\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big]\big)^2}{2\mathrm{C}^n(t,t)}\right\}, \\
&\le \big|\mathrm{E}\big\{\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big\}\big| + \sqrt{2\mathrm{C}^1(t,t)} \qquad (21)
\end{aligned}$$

where Φ is the cdf of the standard normal distribution. The second equality uses the fact that the argument is normally distributed, so that its absolute value follows a folded normal distribution (see, for example, Leone et al., 1961). The inequality $\mathrm{C}^n(t,t) \le \mathrm{C}^1(t,t)$ follows from Lemma 8.5.

Next we want to bound the first term of (21). For $1 \le n < N$, the nth step probabilistic solution is a Gaussian process that is mean-square differentiable on [a,b]. Therefore we can write the difference between the expected probabilistic solution and the exact solution, using a Taylor expansions around s_n,

$$\begin{aligned}
&\mathrm{E}\big[\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big] \\
&= \mathrm{E}\big[\mathrm{u}(s_n) - \mathrm{u}^*(s_n) \mid \mathbf{f}_{1:n}, \theta, \Psi\big] + (t - s_n)\mathrm{E}\big[\mathrm{u}_t(s_n) - f\big(s_n, \mathrm{u}^*(s_n), \theta\big) \mid \mathbf{f}_{1:n}, \theta, \Psi\big] + O(h^2)
\end{aligned}$$

Next we use the Bayesian updating result from Lemma 8.3 to rewrite the probabilistic solution at step n in terms of the probabilistic solution at step $(n-1)$ and the step-ahead derivative realisation $f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big)$, and then rearrange the terms, as follows,

$$\begin{aligned}
&\mathrm{E}\big[\mathrm{u}(t) - \mathrm{u}^*(t) \mid \mathbf{f}_{1:n}, \theta, \Psi\big] \\
&= \mathrm{E}\big[\mathrm{u}(s_n) - \mathrm{u}^*(s_n) \mid \mathbf{f}_{1:n-1}, \theta, \Psi\big] + \frac{\mathrm{C}^{n-1}(s_n, s_n)}{\mathrm{C}_t^{n-1}(s_n, s_n)}\mathrm{E}\big[\big\{f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big) - \mathrm{u}_t(s_n)\big\} \mid \mathbf{f}_{1:n-1}, \theta, \Psi\big] \\
&\quad + (t - s_n)\mathrm{E}\big[f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big) - f\big(s_n, \mathrm{u}^*(s_n), \theta\big) \mid \mathbf{f}_{1:n-1}, \theta, \Psi\big] + O(h^2). \qquad (22)
\end{aligned}$$

Now, we can bound our expected difference between the probabilistic and exact solutions by using (21), (22), and Jensen's inequality:

$$\begin{aligned}
\beta_n(t) \le\ &\beta_{n-1}(s_n) + |t - s_n| \cdot \mathrm{E}\big\{|f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big) - f\big(s_n, \mathrm{u}^*(s_n), \theta\big)| \;\big|\; \mathbf{f}_{1:n-1}, \theta, \Psi\big\} \\
&+ \frac{\mathrm{C}^{n-1}(s_n, s_n)}{\mathrm{C}_t^{n-1}(s_n, s_n)}\ \mathrm{E}\big\{|f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big) - \mathrm{u}_t(s_n)| \;\big|\; \mathbf{f}_{1:n-1}, \theta, \Psi\big\} + \sqrt{2\mathrm{C}^1(t,t)} + O(h^2).
\end{aligned}$$

The Lipschitz continuity of f and boundedness of $\mathrm{E}\big\{|f\big(s_n, \mathrm{u}^{n-1}(s_n), \theta\big) - \mathrm{u}_t(s_n)| \;\big|\; \mathbf{f}_{1:n-1}, \theta, \Psi\big\}$ (by Lipschitz continuity of f and recursive definition of the posterior mean) then implies:

$$\begin{aligned}
\beta_n(t) &\le \beta_{n-1}(s_n) + L|t - s_n|\beta_{n-1}(s_n) + O\left(\frac{\mathrm{C}^{n-1}(s_n, s_n)}{\mathrm{C}_t^{n-1}(s_n, s_n)}\right) + O(\sqrt{\mathrm{C}^1(t,t)}) + O(h^2) \\
&= \beta_{n-1}(s_n)\,(1 + L|t - s_n|) + O(h^2).
\end{aligned}$$

It can be shown (see, for example, Butcher, 2008, p.67-68) that the following inequality holds:

$$\beta_n(t) \le \begin{cases} \beta_0(s_1) + hB(t-a), & L = 0, \\ \exp\{(t-a)L\}\beta_0(s_1) + \exp\{(t-a)L - 1\}hB/L, & L > 0, \end{cases}$$

where B is the constant upper bound on all the remainders. This expression tends to 0 as $\alpha^{-1}, \lambda, h \to 0$, since $\beta_0(s_1) = 0$. Then, taking the expectation of $\beta_n(t)$ with respect to the vector of model evaluations, we obtain the expectation,

$$\mathrm{E}\big\{|\mathrm{u}(t) - \mathrm{u}^*(t)| \;\big|\; \theta, \Psi\big\} \to 0, \quad \text{as } \alpha^{-1}, \lambda, h \to 0.$$

Therefore the probabilistic solution $[\mathrm{u}(t) \mid \theta, \Psi]$ is consistent for $\mathrm{u}^*(t)$. □

Note that the assumption that auxiliary parameters, λ and α^{-1}, associated with the solver tend to zero with the step size is analogous to maintaining a constant number of steps in a k-step numerical method regardless of the step size.

8.7 Covariance properties

In this section we present some results regarding the sequentially obtained covariances, used in the proof of Theorem 8.4.

Lemma 8.5. *For $1 < n \leq N$ and $t \in [a, b]$, the variances for the state and derivative obtained sequentially via Algorithm 1 satisfy:*

$$\begin{aligned} C^n(t,t) &\leq C^1(t,t), \\ C^n_t(t,t) &\leq C^1_t(t,t). \end{aligned}$$

Proof. We use the fact that $\mathrm{C}^n_t(t,t) \geq 0$ for all n and the recurrence from Lemma 8.3, we obtain:

$$\mathrm{C}^n_t(t,t) = \mathrm{C}^{n-1}_t(t,t) - \frac{(\mathrm{C}^{n-1}_t(t,s_n))^2}{\Lambda^{(n,n)}_{n\times n} + \mathrm{C}^{n-1}_t(s_n,s_n)} \leq \mathrm{C}^{n-1}_t(t,t) \leq \cdots \leq \mathrm{C}^1_t(t,t).$$

Similarly,

$$\mathrm{C}^n(t,t) = \mathrm{C}^{n-1}(t,t) - \frac{(\mathrm{C}^{n-1}(t,s_n))^2}{\Lambda^{(n,n)}_{n\times n} + \mathrm{C}^{n-1}_t(s_n,s_n)} \leq \mathrm{C}^{n-1}(t,t) \leq \cdots \leq \mathrm{C}^1(t,t).$$

□

Lemma 8.6. *The covariances, $C^n(t_1,t_2)$ and $C^n_t(t_1,t_2)$, obtained sequentially via Algorithm 1 tend to zero at the rate $O(h^4)$, as $h \to 0$ and $\lambda, \alpha^{-1} = O(h)$ if the covariance function R_λ is stationary and satisfies:*

$$RR(t,t) - RR(t+d,t)RR(t+d,t)/RR(t,t) = O(h^4), \quad \lambda,\ \alpha^{-1} = O(h),\ h \to 0, \tag{23}$$

$$QQ(t,t) - QR(t+d,t)QR^{\dagger}(t+d,t)/RR(t,t) = O(h^4), \quad \lambda,\ \alpha^{-1} = O(h),\ h \to 0, \tag{24}$$

where $d > 0$, $t, t+d \in [a,b]$, and $\lambda \geq h$.

Proof. From Lemma 8.5 and assumption (23) we obtain,

$$\mathrm{C}^n_t(t,t) \leq \mathrm{C}^1_t(t,t) = \mathrm{RR}(t,t) - \mathrm{RR}(t,s_1)\mathrm{RR}(s_1,t)/\mathrm{RR}(s_1,s_1) = O(h^4), \quad \lambda, \alpha^{-1} = O(h),\ h \to 0,\ t \in [a,b],\ 1 \leq n \leq N.$$

Similarly, using Lemma 8.5 and assumption (24) yields,

$$\mathrm{C}^n(t,t) \leq \mathrm{C}^1(t,t) = \mathrm{QQ}(t,t) - \mathrm{QR}(t,s_1)\mathrm{QR}^{\dagger}(t,s_1)/\mathrm{RR}(s_1,s_1) = O(h^4), \quad \lambda, \alpha^{-1} = O(h),\ h \to 0,\ t \in [a,b],\ 1 \leq n \leq N.$$

Then, by the Cauchy-Schwarz inequality,

$$\left|\mathrm{C}^n_t(t_1,t_2)\right|, \left|\mathrm{C}^n(t_1,t_2)\right| = O(h^4), \quad \lambda, \alpha^{-1} = O(h),\ h \to 0,\ t_1, t_2 \in [a,b],\ 1 \leq n \leq N.$$

□

It is straightforward to show that the square exponential and uniform covariance functions considered in this paper are stationary and symmetric and satisfy conditions (23) and (24).

Lemma 8.7. *The model-prior mismatch covariance matrix, $\Lambda_{n\times n} := diag\{0, C^1_t(s_2), \ldots, C^{n-1}_t(s_n)\}$, obtained via Algorithm 1 tends to zero as $h \to 0$ at the rate $O(h^4)$ when conditions (23) and (24) are satisfied.*

Proof. The proof follows immediately from Lemma 8.6 and the definition of $\Lambda_{n\times n}$. □