Teymur, Onur, Zygalakis, Konstantinos and Calderhead, Ben (2016) *Probabilistic Linear Multistep Methods.* Advances in Neural Information Processing Systems 29 . pp. 4321-4328.

Probabilistic Linear Multistep Methods

Onur Teymur
Department of Mathematics
Imperial College London
o@teymur.uk

Konstantinos Zygalakis
School of Mathematics
University of Edinburgh
k.zygalakis@ed.ac.uk

Ben Calderhead
Department of Mathematics
Imperial College London
b.calderhead@imperial.ac.uk

Abstract

We present a derivation and theoretical investigation of the Adams-Bashforth and Adams-Moulton family of linear multistep methods for solving ordinary differential equations, starting from a Gaussian process (GP) framework. In the limit, this formulation coincides with the classical deterministic methods, which have been used as higher-order initial value problem solvers for over a century. Furthermore, the natural probabilistic framework provided by the GP formulation allows us to derive *probabilistic* versions of these methods, in the spirit of a number of other probabilistic ODE solvers presented in the recent literature [1, 2, 3, 4]. In contrast to higher-order Runge-Kutta methods, which require multiple intermediate function evaluations per step, Adams family methods make use of previous function evaluations, so that increased accuracy arising from a higher-order multistep approach comes at very little additional computational cost. We show that through a careful choice of covariance function for the GP, the posterior mean and standard deviation over the numerical solution can be made to exactly coincide with the value given by the deterministic method and its local truncation error respectively. We provide a rigorous proof of the convergence of these new methods, as well as an empirical investigation (up to fifth order) demonstrating their convergence rates in practice.

1 Introduction

Numerical solvers for differential equations are essential tools in almost all disciplines of applied mathematics, due to the ubiquity of real-world phenomena described by such equations, and the lack of exact solutions to all but the most trivial examples. The performance – speed, accuracy, stability, robustness – of the numerical solver is of great relevance to the practitioner. This is particularly the case if the computational cost of accurate solutions is significant, either because of high model complexity or because a high number of repeated evaluations are required (which is typical if an ODE model is used as part of a statistical inference procedure, for example). A field of work has emerged which seeks to quantify this performance – or indeed lack of it – by modelling the numerical errors *probabilistically*, and thence trace the effect of the chosen numerical solver through the entire computational pipeline [5]. The aim is to be able to make meaningful quantitative statements about the uncertainty present in the resulting scientific or statistical conclusions.

Recent work in this area has resulted in the development of probabilistic numerical methods, first conceived in a very general way in [6]. An recent summary of the state of the field is given in [7]. The particular case of ODE solvers was first addressed in [8], formalised and extended in [1, 2, 3] with a number of theoretical results recently given in [4]. The present paper modifies and extends the constructions in [1, 4] to the multistep case, improving the order of convergence of the method but avoiding the simplifying linearisation of the model required by the approaches of [2, 3]. Furthermore we offer extensions to the convergence results in [4] to our proposed method and give empirical results confirming convergence rates which point to the practical usefulness of our higher-order approach without significantly increasing computational cost.

1.1 Mathematical setup

We consider an Initial Value Problem (IVP) defined by an ODE

$$\frac{\mathrm{d}}{\mathrm{d}t}y(t,\theta) = f(y(t,\theta),t), \qquad y(t_0,\theta) = y_0 \qquad (1)$$

Here $y(\cdot,\theta): \mathbb{R}^+ \to \mathbb{R}^d$ is the solution function, $f: \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}^d$ is the vector-valued function that defines the ODE, and $y_0 \in \mathbb{R}^d$ is a given vector called the initial value. The dependence of y on an m-dimensional parameter $\theta \in \mathbb{R}^m$ will be relevant if the aim is to incorporate the ODE into an inverse problem framework, and this parameter is of scientific interest. Bayesian inference under this setup (see [9]) is covered in most of the other treatments of this topic but is not the main focus of this paper; we therefore suppress θ for the sake of clarity.

Some technical conditions are required in order to justify the existence and uniqueness of solutions to (1). We assume that f is evaluable point-wise given y and t and also that it satisfies the Lipschitz condition in y, namely $||f(y_1,t) - f(y_2,t)|| \leq L_f||y_1 - y_2||$ for some $L_f \in \mathbb{R}^+$ and all t, y_1 and y_2; and also is continuous in t. These conditions imply the existence of a unique solution, by a classic result usually known as the Picard-Lindelöf Theorem [10].

We consider a finite-dimensional discretisation of the problem, with our aim being to numerically generate an N-dimensional vector[1] $y_{1:N}$ approximating the true solution $y(t_{1:N})$ in an appropriate sense. Following [1], we consider the joint distribution of $y_{1:N}$ and the auxiliary variables $f_{0:N}$ (obtained by evaluating the function f), with each y_i obtained by sequentially conditioning on previous evaluations of f. A basic requirement is that the marginal mean of $y_{1:N}$ should correspond to some deterministic iterative numerical method operating on the grid $t_{1:N}$. In our case this will be a linear multistep method (LMM) of specified type. [2]

Firstly we telescopically factorise the joint distribution as follows:

$$p(y_{1:N}, f_{0:N}|y_0) = p(f_0|y_0) \prod_{i=0}^{N-1} p(y_{i+1}|y_{0:i}, f_{0:i}) \, p(f_{i+1}|y_{0:i+1}, f_{0:i}) \qquad (2)$$

We can now make simplifying assumptions about the constituent distributions. Firstly since we have assumed that f is evaluable point-wise given y and t,

$$p(f_i|y_i, \dots) = p(f_i|y_i) = \delta_{f_i}\big(f(y_i,t_i)\big), \qquad (3)$$

which is a Dirac-delta measure equivalent to simply performing this evaluation deterministically. Secondly, we assume a finite moving window of dependence for each new state – in other words y_{i+1} is only allowed to depend on y_i and $f_i, f_{i-1}, \dots, f_{i-(s-1)}$ for some $s \in \mathbb{N}$. This corresponds to the inputs used at each iteration of the s-step Adams-Bashforth method. For $i < s$ we will assume dependence on only those derivative evaluations up to i; this initialisation detail is discussed briefly in Section 4. Strictly speaking, f_N is superfluous to our requirements (since we already have y_N) and thus we can rewrite (2) as

$$p(y_{1:N}, f_{0:N-1}|y_0) = \prod_{i=0}^{N-1} p(f_i|y_i) \, p(y_{i+1}|y_i, f_{\max(0,i-s+1):i}) \qquad (4)$$

$$= \prod_{i=0}^{N-1} \delta_{f_i}(f(y_i,t_i)) \underbrace{p(y_{i+1}|y_i, f_{\max(0,i-s+1):i})}_{*} \qquad (5)$$

The conditional distributions $*$ are the primary objects of our study – we will define them by constructing a particular Gaussian process prior over all variables, then identifying the appropriate (Gaussian) conditional distribution. Note that a simple modification to the decomposition (2) allows the same set-up to generate an $(s+1)$-step Adams-Moulton iterator[3] – the implicit multistep method where y_{i+1} depends in addition on f_{i+1}. At various stages of this paper this extension is noted but omitted for reasons of space – the collected results are given in Appendix C.

---

[1]The notation $y_{0:N}$ denotes the vector $(y_0, \dots, y_N)$, and analogously $t_{0:N}$, $f_{0:N}$ etc.

[2]We argue that the connection to some specific deterministic method is a desirable feature, since it aids interpretability and allows much of the well-developed theory of IVP solvers to be inherited by the probabilistic solver. This is a particular strength of the formulation in [4] which was lacking in all previous works.

[3]The convention is that the number of steps is equal to the total number of derivative evaluations used in each iteration, hence the s-step AB and $(s+1)$-step AM methods both go 'equally far back'.

2

Linear multistep methods

We give a very short summary of Adams family LMMs and their conventional derivation via interpolating polynomials. For a fuller treatment of this well-studied topic we refer the reader to the comprehensive references [10, 11, 12]. Using the usual notation we write y_i for the numerical estimate of the true solution $y(t_i)$, and f_i for the estimate of $f(t_i) \equiv y'(t_i)$.

The classic s-step Adams-Bashforth method calculates y_{i+1} by constructing the unique polynomial $P_i(\omega) \in \mathbb{P}_{s-1}$ interpolating the points $\{f_{i-j}\}_{j=0}^{s-1}$. This is given by Lagrange's method as

$$P_i(\omega) = \sum_{j=0}^{s-1} \ell_j^{0:s-1}(\omega) f_{i-j} \qquad \ell_j^{0:s-1}(\omega) = \prod_{\substack{k=0 \\ k \neq j}}^{s-1} \frac{\omega - t_{i-k}}{t_{i-j} - t_{i-k}} \tag{6}$$

The $\ell_j^{0:s-1}(\omega)$ are known as Lagrange polynomials, have the property that $\ell_p^{0:s-1}(t_{i-q}) = \delta_{pq}$, and form a basis for the space $\mathbb{P}_{s-1}$ known as the Lagrange basis. The Adams-Bashforth iteration then proceeds by writing the integral version of (1) as $y(t_{i+1}) - y(t_i) \equiv \int_{t_i}^{t_{i+1}} f(y, t) \, \mathrm{d}t$ and approximating the function under the integral by the extrapolated interpolating polynomial to give

$$y_{i+1} - y_i \approx \int_{t_i}^{t_{i+1}} P_i(\omega) \, \mathrm{d}\omega = h \sum_{j=0}^{s-1} \beta_{j,s}^{AB} f_{i-j} \tag{7}$$

where $h = t_{i+1} - t_i$ and the $\beta_{j,s}^{AB} \equiv h^{-1} \int_0^h \ell_j^{0:s-1}(\omega) \, \mathrm{d}\omega$ are the Adams-Bashforth coefficients for order s, all independent of h and summing to 1. Note that if f is a polynomial of degree $s-1$ (so $y(t)$ is a polynomial of degree s) this procedure will give the next solution value *exactly*. Otherwise the extrapolation error in f_{i+1} is of order $O(h^s)$ and in y_{i+1} (after an integration) is of order $O(h^{s+1})$. So the local truncation error is $O(h^{s+1})$ and the global error $O(h^s)$ [10].

Adams-Moulton methods are similar except that the polynomial $Q_i(\omega) \in \mathbb{P}_s$ interpolates the $s+1$ points $\{f_{i-j}\}_{j=-1}^{s-1}$. The resulting equation analogous to (7) is thus an implicit one, with the unknown y_{i+1} appearing on both sides. Typically AM methods are used in conjunction with an AB method of one order lower, in a 'predictor-corrector' arrangement. Here, a predictor value y_{i+1}^* is calculated using an AB step; this is then used to estimate $f_{i+1}^* = f(y_{i+1}^*)$; and finally an AM step uses this value to calculate y_{i+1}. We again refer the reader to Appendix C for details of the AM construction.

2 Derivation of Adams family LMMs via Gaussian processes

We now consider a formulation of the Adams-Bashforth family starting from a Gaussian process framework and then present a probabilistic extension. We fix a joint Gaussian process prior over $y_{i+1}, y_i, f_i, f_{i-1}, \ldots, f_{i-s+1}$ as follows. We define two vectors of functions $\phi(\omega)$ and $\Phi(\omega)$ in terms of the Lagrange polynomials $\ell_j^{0:s-1}(\omega)$ defined in (6) as

$$\phi(\omega) = \begin{pmatrix} 0 & \ell_0^{0:s-1}(\omega) & \ell_1^{0:s-1}(\omega) & \ldots & \ell_{s-1}^{0:s-1}(\omega) \end{pmatrix}^T \tag{8}$$

$$\Phi(\omega) = \int \phi(\omega) \, \mathrm{d}\omega = \begin{pmatrix} 1 & \int \ell_0^{0:s-1}(\omega) \, \mathrm{d}\omega & \ldots & \int \ell_{s-1}^{0:s-1}(\omega) \, \mathrm{d}\omega \end{pmatrix}^T \tag{9}$$

The elements (excluding the first) of $\phi(\omega)$ form a basis for $\mathbb{P}_{s-1}$ and the elements of $\Phi(\omega)$ form a basis for $\mathbb{P}_s$. The initial 0 in $\phi(\omega)$ is necessary to make the dimensions of the two vectors equal, so we can correctly define products such as $\Phi(\omega)^T \phi(\omega)$ which will be required later. The first element of $\Phi(\omega)$ can be any non-zero constant C; the analysis later is unchanged and we therefore take $C = 1$.

Since we will solely be interested in values of the argument ω corresponding to discrete equispaced time-steps $t_j - t_{j-1} = h$ indexed relative to the current time-point $t_i = 0$, we will make our notation more concise by writing ϕ_{i+k} for $\phi(t_{i+k})$, and similarly Φ_{i+k} for $\Phi(t_{i+k})$. We now use these vectors of basis functions to define a joint Gaussian process prior as follows:

$$
\begin{pmatrix} y_{i+1} \\ y_i \\ f_i \\ f_{i-1} \\ \vdots \\ f_{i-s+1} \end{pmatrix} = \mathcal{N} \left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \Phi_{i+1}^T \Phi_{i+1} & \Phi_{i+1}^T \Phi_i & \Phi_{i+1}^T \phi_i & \cdots & \Phi_{i+1}^T \phi_{i-s+1} \\ \Phi_i^T \Phi_{i+1} & \Phi_i^T \Phi_i & \Phi_i^T \phi_i & \cdots & \Phi_i^T \phi_{i-s+1} \\ \phi_i^T \Phi_{i+1} & \phi_i^T \Phi_i & \phi_i^T \phi_i & \cdots & \phi_i^T \phi_{i-s+1} \\ \phi_{i-1}^T \Phi_{i+1} & \phi_{i-1}^T \Phi_i & \phi_{i-1}^T \phi_i & \cdots & \phi_{i-1}^T \phi_{i-s+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{i-s+1}^T \Phi_{i+1} & \phi_{i-s+1}^T \Phi_i & \phi_{i-s+1}^T \phi_i & \cdots & \phi_{i-s+1}^T \phi_{i-s+1} \end{pmatrix} \right]
$$
$$(10)$$

This construction works because $y' = f$ and differentiation is a linear operator; the rules for the transformation of the covariance elements is given in Section 9.4 of [13] and can easily be seen to correspond to the defined relationship between $\phi(\omega)$ and $\Phi(\omega)$.

Recalling the decomposition in (5), we are interested in the conditional distribution $p(y_{i+1}|y_i, f_{i-s+1:i})$. This is also Gaussian, with mean and covariance given by the standard formulae for Gaussian conditioning. This construction now allows us to state the following result:

Proposition 1. *The conditional distribution $p(y_{i+1}|y_i, f_{i-s+1:i})$ under the Gaussian process prior given in (10), with covariance kernel basis functions as in (8) and (9), is a δ-measure concentrated on the s-step Adams-Bashforth predictor $y_i + h \sum_{j=0}^{s-1} \beta_{j,s}^{AB} f_{i-j}$.*

The proof of this proposition is given in Appendix A.

Because of the natural probabilistic structure provided by the Gaussian process framework, we can augment the basis function vectors $\phi(\omega)$ and $\Phi(\omega)$ to generate a conditional distribution for y_{i+1} that has non-zero variance. By choosing a particular form for this augmented basis we can obtain an expression for the standard deviation of y_{i+1} that is exactly equal to the leading-order local truncation error of the corresponding deterministic method.

We will expand the vectors $\phi(\omega)$ and $\Phi(\omega)$ by one component, chosen so that the new vector comprises elements that span a polynomial space of order one greater than before. Define the augmented bases $\phi^+(\omega)$ and $\Phi^+(\omega)$ as

$$
\phi(\omega)^+ = \begin{pmatrix} 0 & \ell_0^{0:s-1}(\omega) & \ell_1^{0:s-1}(\omega) & \cdots & \ell_{s-1}^{0:s-1}(\omega) & \alpha h^s \ell_{-1}^{-1:s-1}(\omega) \end{pmatrix}^T
\tag{11}
$$

$$
\Phi(\omega)^+ = \begin{pmatrix} 1 & \int \ell_0^{0:s-1}(\omega) \, d\omega & \cdots & \int \ell_{s-1}^{0:s-1}(\omega) \, d\omega & \int \alpha h^s \ell_{-1}^{-1:s-1}(\omega) \, d\omega \end{pmatrix}^T
\tag{12}
$$

The additional term at the end of $\phi^+(\omega)$ is the polynomial of order s which arises from interpolating f at $s+1$ points (with the additional point at t_{i+1}) and choosing the basis function corresponding to the root at t_{i+1}, scaled by αh^s with α a positive constant whose role will be explained in the next section. The elements of these vectors span $\mathbb{P}_s$ and $\mathbb{P}_{s+1}$ respectively. With this new basis we can give the following result:

Proposition 2. *The conditional distribution $p(y_{i+1}|y_i, f_{i-s+1:i})$ under the Gaussian process prior given in (10), with covariance kernel basis functions as in (11) and (12), is Gaussian with mean equal to the s-step Adams-Bashforth predictor $y_i + h \sum_{j=0}^{s-1} \beta_{j,s}^{AB} f_{i-j}$ and, setting $\alpha = y^{(s+1)}(\eta)$ for some $\eta \in (t_{i-s+1}, t_{i+1})$, standard deviation equal to its local truncation error.*

The proof is given in Appendix B. In order to de-mystify the construction, we now exhibit a concrete example for the case $s = 3$. The conditional distribution of interest is $p(y_{i+1}|y_i, f_i, f_{i-1}, f_{i-2}) \equiv p(y_{i+1}|y_i, f_{i:i-2})$. In the deterministic case, the vectors of basis functions become

$$
\phi(\omega)_{s=3} = \begin{pmatrix} 0 & \dfrac{(\omega + h)(\omega + 2h)}{2h^2} & \dfrac{\omega(\omega + 2h)}{-h^2} & \dfrac{\omega(\omega + h)}{2h^2} \end{pmatrix}
$$

$$
\Phi(\omega)_{s=3} = \begin{pmatrix} 1 & \dfrac{\omega(2\omega^2 + 9h\omega + h^2)}{12h^2} & \dfrac{\omega^2(\omega + 3h)}{-3h^2} & \dfrac{\omega^2(2\omega + 3h)}{12h^2} \end{pmatrix}
$$

4

and simple calculations give that

$$\mathbb{E}(y_{i+1}|y_i, f_{i:i-2}) = y_i + h\left(\frac{23}{12}f_i - \frac{4}{3}f_{i-1} + \frac{5}{12}f_{i-2}\right) \qquad \mathrm{Var}(y_{i+1}|y_i, f_{i:i-2}) = 0$$

The probabilistic version follows by setting

$$\phi^+(\omega)_{s=3} = \left(0 \quad \frac{(\omega+h)(\omega+2h)}{2h^2} \quad \frac{\omega(\omega+2h)}{-h^2} \quad \frac{\omega(\omega+h)}{2h^2} \quad \frac{\alpha\omega(\omega+h)(\omega+2h)}{6}\right)$$

$$\Phi^+(\omega)_{s=3} = \left(1 \quad \frac{\omega\left(2\omega^2+9h\omega+h^2\right)}{12h^2} \quad \frac{\omega^2\left(x+3h\right)}{-3h^2} \quad \frac{\omega^2\left(2\omega+3h\right)}{12h^2} \quad \frac{\alpha\omega^2(\omega+2h)^2}{24}\right)$$

and further calculation shows that

$$\mathbb{E}(y_{i+1}|y_i, f_{i:i-2}) = y_i + h\left(\frac{23}{12}f_i - \frac{4}{3}f_{i-1} + \frac{5}{12}f_{i-2}\right) \qquad \mathrm{Var}(y_{i+1}|y_i, f_{i:i-2}) = \left(\frac{3h^4\alpha}{8}\right)^2$$

An entirely analogous argument can be shown to reproduce and probabilistically extend the implicit Adams-Moulton scheme. The Gaussian process prior now includes f_{i+1} as an additional variable and the correlation structure and vectors of basis functions are modified accordingly. The required modifications are given in Appendix C and a explicit derivation for the 4-step AM method is given in Appendix D.

2.1 The role of α

Replacing α in (11) by $y^{(s+1)}(\eta)$, with $\eta \in (t_{i-s+1}, t_{i+1})$, makes the variance of the integrator coincide exactly with the local truncation error of the underlying deterministic method.[4]

This is of course of limited utility unless higher derivatives of $y(t)$ are available, and even if they are, η is itself unknowable in general. However it is possible to estimate the integrator variance in a systematic way by using backward difference approximations [14] to the required derivative at t_{i+1}. We show this by expanding the s-step Adams-Bashforth iterator as

$$
\begin{aligned}
y_{i+1} &= y_i + h\sum_{j=0}^{s-1}\beta_{j,s}^{AB}f_{i-j} + h^{s+1}C_s^{AB}y^{(s+1)}(\eta) & \eta \in [t_{i-s+1}, t_{i+1}]\\
&= y_i + h\sum_{j=0}^{s-1}\beta_{j,s}^{AB}f_{i-j} + h^{s+1}C_s^{AB}y^{(s+1)}(t_{i+1}) + O(h^{s+2})\\
&= y_i + h\sum_{j=0}^{s-1}\beta_{j,s}^{AB}f_{i-j} + h^{s+1}C_s^{AB}f^{(s)}(t_{i+1}) + O(h^{s+2}) & \text{since } y' = f\\
&= y_i + h\sum_{j=0}^{s-1}\beta_{j,s}^{AB}f_{i-j} + h^{s+1}C_s^{AB}\left[h^{-s}\sum_{k=0}^{s-1+p}\delta_{k,s-1+p}f_{i-k} + O(h^p)\right] + O(h^{s+2})\\
&= y_i + h\sum_{j=0}^{s-1}\beta_{j,s}^{AB}f_{i-j} + hC_s^{AB}\sum_{k=0}^{s}\delta_{k,s}f_{i-k} + O(h^{s+2}) & \text{if we set } p = 1 \quad (13)
\end{aligned}
$$

where $\beta_{\cdot,s}^{AB}$ are the set of coefficients and C_s^{AB} the local truncation error constant for the s-step Adams-Bashforth method, and $\delta_{\cdot,s-1+p}$ are the set of backward difference coefficients for estimating the sth derivative of f to order $O(h^p)$ [14].

In other words, the constant α can be substituted with $h^{-s}\sum_{k=0}^{s}\delta_{k,s}f_{i-k}$, using already available function values and to adequate order. It is worth noting that collecting the coefficients $\beta_{\cdot,s}^{AB}$ and $\delta_{\cdot,s}$ results in an expression equivalent to the Adams-Bashforth method of order $s+1$ and therefore, this procedure is in effect employing two integrators of different orders and estimating the truncation error from the difference of the two.[5] This principle is similar to the classical Milne Device [12], which pairs an AB and and AM iterator to achieve the same thing. Using the Milne Device to generate a value for the error variance is also straightforward within our framework, but requires two evaluations of f at each iteration (one of which immediately goes to waste) instead of the approach presented here, which only requires one.

---

[4]We do not claim that this is the only possible way of modelling the numerical error in the solver. The question of how to do this accurately is an open problem in general, and is particularly challenging in the multi-dimensional case. In many real world problems different noise scales will be appropriate for different dimensions and – especially in 'hierarchical' models arising from higher-order ODEs – non-Gaussian noise is to be expected. That said, the Gaussian assumption as a first order approximation for numerical error is present in virtually all work on this subject and goes all the way back to [8]. We adopt this premise throughout, whilst noting this interesting unresolved issue.

[5]An explicit derivation of this for $s = 3$ is given in Appendix E.

3 Convergence of the probabilistic Adams-Bashforth integrator

We now give the main result of our paper, which demonstrates that the convergence properties of the probabilistic Adams-Bashforth integrator match those of its deterministic counterpart.

Theorem 3. *Consider the s-step deterministic Adams-Bashforth integrator given in Proposition 1, which is of order s. Then the probabilistic integrator constructed in Proposition 2 has the same mean square error as its deterministic counterpart. In particular*

$$\max_{0 \leq kh \leq T} \mathbb{E}|Y_k - y_k|^2 \leq Kh^{2s}$$

where $Y_k \equiv y(t_k)$ denotes the true solution, y_k the numerical solution, and K is a positive real number depending on T but independent of h.

The proof of this theorem is given in Appendix F, and follows a similar line of reasoning to that given for a one-step probabilistic Euler integrator in [4]. In particular, we deduce the convergence of the algorithm by extrapolating from the local error. The additional complexity arises due to the presence of the stochastic part, which means we cannot rely directly on the theory of difference equations and the representations of their solutions. Instead, following [15], we rewrite the defining s-step recurrence equation as a one-step recurrence equation in a higher dimensional space.

4 Implementation

We now have an implementable algorithm for an s-step probabilistic Adams-Bashforth integrator. Firstly, an accurate initialisation is required for the first s iterations – this can be achieved with, for example, a Runge-Kutta method of sufficiently high order.[6] Secondly, at iteration i, the preceding s stored function evaluations are used to find the posterior mean and variance of y_{i+1}. The integrator then advances by generating a realisation of the posterior measure derived in Proposition 2. Following [1], a Monte Carlo repetition of this procedure with different random seeds can then be used as an effective way of generating propagated uncertainty estimates at any time $0 < T < \infty$.

4.1 Example – Chua circuit

The Chua circuit [16] is the simplest electronic circuit that exhibits chaotic behaviour, and has been the subject of extensive study – in both the mathematics and electronics communities – for over 30 years. Readers interested in this rich topic are directed to [17] and the references therein. The defining characteristic of chaotic systems is their unpredictable long-term sensitivity to tiny changes in initial conditions, which also manifests itself in the sudden amplification of error introduced by any numerical scheme. It is therefore of interest to understand the limitations of a given numerical method applied to such a problem – namely the point at which the solution can no longer be taken to be a meaningful approximation of the ground truth. Probabilistic integrators allow us to do this in a natural way [1].

The Chua system is given by $x' = \alpha(y - (1 + h_1)x - h_3 x^3)$, $y' = x - y + z$, $z' = -\beta y - \gamma z$. We use parameter values $\alpha = -1.4157$, $\beta = 0.02944201$, $\gamma = 0.322673579$, $h_1 = -0.0197557699$, $h_3 = -0.0609273571$ and initial conditions $x_0 = 0$, $y_0 = 0.003$, $z_0 = 0.005$. This particular choice is taken from 'Attractor CE96' in [18]. Using the probabilistic version of the Adams-Bashforth integrator with $s > 1$, it is possible to delay the point at which numerical path diverges from the truth, *with effectively no additional evaluations of f required* compared to the one-step method. This is demonstrated in Figure 1. Our approach is therefore able to combine the benefits of classical higher-order methods with the additional insight into solution uncertainty provided by a probabilistic method.

4.2 Example – Lotka-Volterra model

We now apply the probabilistic integrator to a simple periodic predator-prey model given by the system $x' = \alpha x - \beta xy$, $y' = \gamma xy - \delta y$ for parameters $\alpha = 1$, $\beta = 0.3$, $\gamma = 1$ and $\delta = 0.7$. We demonstrate the convergence behaviour stated in Theorem 3 empirically.

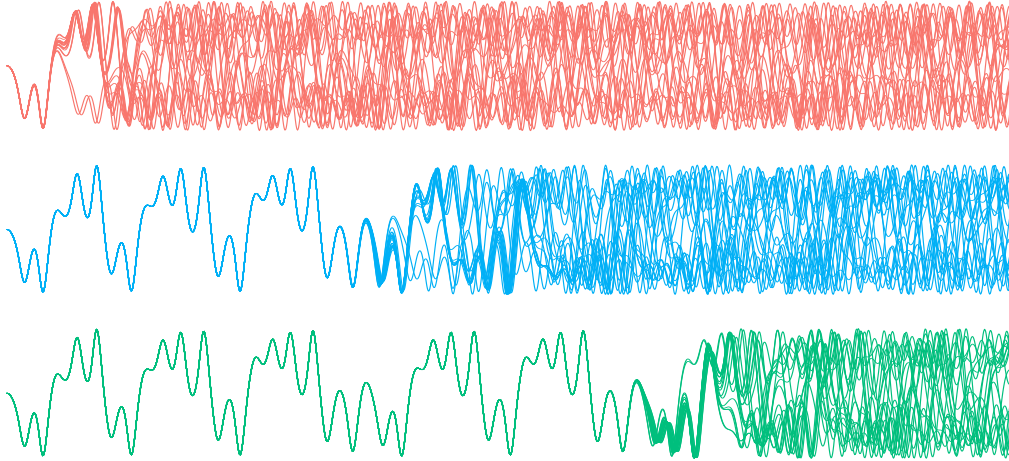[6]We use a (packaged) adaptive Runge-Kutta-Fehlberg solver of 7th order with 8th order error control.

Figure 1: Time series for the x-component in the Chua circuit model described in Section 4.1, solved 20 times for $0 \leq t \leq 1000$ using an s-step probabilistic AB integrator with $s = 1$ (top), $s = 3$ (middle), $s = 5$ (bottom). Step-size remains $h = 0.01$ throughout. Wall-clock time for each simulation was close to constant (± 10 per cent – the difference primarily accounted for by the RKF initialisation procedure).

The left-hand plot in Figure 2 shows the sample mean of the absolute error of 200 realisations of the probabilistic integrator plotted against step-size, on a log-log scale. The differing orders of convergence of the probabilistic integrators are easily deduced from the slopes of the lines shown.

The right-hand plot shows the actual error value (no logarithm or absolute value taken) of the same 200 realisations, plotted individually against step-size. This plot shows that the error in the one-step integrator is consistently positive, whereas for two- and three-step integrators is approximately centred around 0. (This is also visible with the same data if the plot is zoomed to more closely examine the range with small h.) Though this phenomenon can be expected to be somewhat problem-dependent, it is certainly an interesting observation which may have implications for bias reduction in a Bayesian inverse problem setting.
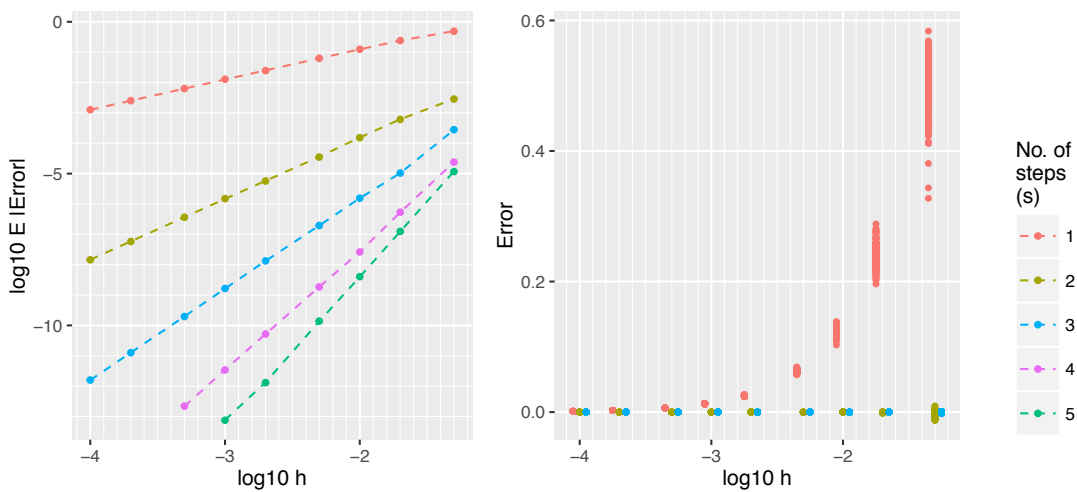

Figure 2: Empirical error analysis for the x-component of 200 realisations of the probabilistic AB integrator as applied to the Lotka-Volterra model described in Section 4.2. The left-hand plot shows the convergence rates for AB integrators of orders 1-5, while the right-hand plot shows the distribution of error around zero for integrators of orders 1-3.

7

5   Conclusion

We have given a derivation of the Adams-Bashforth and Adams-Moulton families of linear multistep ODE integrators, making use of a Gaussian process framework, which we then extend to develop their probabilistic counterparts.

We have shown that the derived family of probabilistic integrators result in a posterior mean at each step that exactly coincides with the corresponding deterministic integrator, with the posterior standard deviation equal to the deterministic method's local truncation error. We have given the general forms of the construction of these new integrators to arbitrary order. Furthermore, we have investigated their theoretical properties and provided a rigorous proof of their rates of convergence, Finally we have demonstrated the use and computational efficiency of probabilistic Adams-Bashforth methods by implementing the solvers up to fifth order and providing example solutions of a chaotic system, and well as empirically verifying the convergence rates in a Lotka-Voltera model.

We hope the ideas presented here will add to the arsenal of any practitioner who uses numerical methods in their scientific analyses, and contributes a further tool in the emerging field of probabilistic numerical methods.

References

[1]   O. A. CHKREBTII, D. A. CAMPBELL, B. CALDERHEAD, and M. A. GIROLAMI. Bayesian Solution Uncertainty Quantification for Differential Equations. *Bayesian Analysis*, 2016.

[2]   P. HENNIG and S. HAUBERG. Probabilistic Solutions to Differential Equations and their Application to Riemannian Statistics. In, *Proc. of the 17th int. Conf. on Artificial Intelligence and Statistics (AISTATS)*. Vol. 33. JMLR, W&CP, 2014.

[3]   M. SCHOBER, D. K. DUVENAUD, and P. HENNIG. Probabilistic ODE Solvers with Runge-Kutta Means. In Z. GHAHRAMANI, M. WELLING, C. CORTES, N. D. LAWRENCE, and K. Q. WEINBERGER, editors, *Advances in Neural Information Processing Systems 27*, pp. 739–747. Curran Associates, Inc., 2014.

[4]   P. R. CONRAD, M. GIROLAMI, S. SÄRKKÄ, A. STUART, and K. ZYGALAKIS. Statistical Analysis of Differential Equations: Introducing Probability Measures on Numerical Solutions. *Statistics and Computing*, 2016.

[5]   M. C. KENNEDY and A. O'HAGAN. Bayesian Calibration of Computer Models. *Journal of the Royal Statistical Society: Series B*, 63(3):425–464, 2001.

[6]   P. DIACONIS. Bayesian Numerical Analysis. In J. BERGER and S. GUPTA, editors, *Statistical Decision Theory and Related Topics IV*. Vol. 1, pp. 163–175. Springer, 1988.

[7]   P. HENNIG, M. A. OSBORNE, and M. GIROLAMI. Probabilistic Numerics and Uncertainty in Computations. *Proc. R. Soc. A*, 471(2179):20150142, 2015.

[8]   J. SKILLING. Bayesian Numerical Analysis. In J. W. T. GRANDY and P. W. MILONNI, editors, *Physics and Probability*, pp. 207–222. Cambridge University Press, 1993.

[9]   M. GIROLAMI. Bayesian Inference for Differential Equations. *Theor. Comp. Sci.*, 408(1):4–16, 2008.

[10]   A. ISERLES. *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, 2nd ed., 2008.

[11]   E. HAIRER, S. NØRSETT, and G. WANNER. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Of *Springer Series in Computational Mathematics*. Springer, 2008.

[12]   J. BUTCHER. *Numerical Methods for Ordinary Differential Equations: Second Edition*. Wiley, 2008.

[13]   C. RASMUSSEN and C. WILLIAMS. *Gaussian Processes for Machine Learning*. University Press Group Limited, 2006.

[14]   B. FORNBERG. Generation of Finite Difference Formulas on Arbitrarily Spaced Grids. *Mathematics of Computation*, 51(184):699–706, 1988.

[15]   E. BUCKWAR and R. WINKLER. Multistep Methods for SDEs and Their Application to Problems with Small Noise. *SIAM J. Numer. Anal.*, 44(2):779–803, 2006.

[16]   L. O. CHUA. The Genesis of Chua's Circuit. *Archiv für Elektronik und Übertragungstechnik*, 46(4):250–257, 1992.

[17]   L. O. CHUA. Chua Circuit. *Scholarpedia*, 2(10):1488, 2007.

[18]   E. BILOTTA and P. PANTANO. *A Gallery of Chua Attractors*. World Scientific, 2008.