On solving Ordinary Differential Equations using Gaussian Processes

David Barber
Department of Computer Science
University College London

August 17, 2014

Abstract

We describe a set of Gaussian Process based approaches that can be used to solve non-linear Ordinary Differential Equations. We suggest an explicit probabilistic solver and two implicit methods, one analogous to Picard iteration and the other to gradient matching. All methods have greater accuracy than previously suggested Gaussian Process approaches. We also suggest a general approach that can yield error estimates from any standard ODE solver.

1 The Initial Value Problem

Given an Ordinary Differential Equation (ODE) with known initial condition $x(t_1) = x_1$

$$\frac{d}{dt}x(t) = f(t, x(t), \theta) \tag{1}$$

the Initial Value Problem (IVP) is to find the differentiable function $x(t)$ over some specified time interval $t \in [t_1, t_T]$ that satisfies the ODE subject to the initial value condition. In general $x(t)$ is a vector so that higher order scalar ODEs can be embedded as first order vector ODEs [8]. In general this problem requires an approximate numerical solution and we denote the approximation at time t_n to $x(t_n)$ by x_n. There is a vast literature on this topic (see [8] for an introduction) and several families of techniques that can be applied such as one-step methods, multistep methods, fixed and variable step length, and implicit and explicit approaches. In general there is no single 'best' method with the methods having different properties in terms of numerical stability, speed, number of function f evaluations, parallelisabiltiy *etc.*

Recently there has been interest in the machine learning community in the application of Gaussian Processes for the IVP[1] [9, 5, 2] and estimation of ODE parameters given potentially noisy observations $\mathcal{D}$ [1, 3, 10]. An ideal approach to parameter estimation is based on Bayesian Numerical Integration. Writing $t_1, \ldots, t_N$ for the times at which data is observed and $x(t_n)$ for the true solution to the IVP at those times,

$$p(\theta, x_{2:N}|x_1, \mathcal{D}) \propto p(\mathcal{D}|x_{2:N})p(x_{2:N}|x_1, \theta) \tag{2}$$

where the term $p(x_{2:N}|x_1, \theta)$ represents a distribution over true solutions given the initial value. Generally this otherwise ideal approach is problematic since classical IVP solution techniques do not produce a distribution over solutions $x_{2:N}$, meaning that the uncertainty (which must exist due to the numerical approximation) in the solution is not correctly accounted for.

The approaches in [1, 3, 10] use Gaussian Processes to circumvent the requirement to produce $p(x_{2:N}|x_1, \theta)$ and work by implicitly fitting an alternative function to the data whose gradient must match the gradient specified by the ODE at the observation times. Whilst these recent parameter estimation approaches that avoid the requirement to find $p(x_{2:N}|x_1, \theta)$ look promising, it nevertheless remains of interest to find distributions $p(x_{2:N}|x_1, \theta)$ since these can be used to solve the IVP and characterise uncertainty in the solution. This is the focus of this work, in which we assume that the parameters θ of the ODE are known, but an estimate of uncertainty in the solution is required[2].

[1] The more general boundary value problems can also be addressed using related approaches.

[2] We therefore assume θ is fixed and known and drop the notational conditioning on θ throughout.

1.1 Gaussian Processes and Linear ODEs

In the case that f is linear in $x(t)$, the solution involves integrals of matrix exponentials, which generally cannot be computed in closed form but can be approximated using for example the Magnus expansion [8]. An alternative approximate approach that avoids explicit integration and generalises the solution to the case of additive Gaussian noise is to assume that $x(t)$ follows a Gaussian Process (GP) with covariance function $C(t, t')$, see for example [4]. Then writing $f(t, x(t))$ in terms of a matrix L, time-varying term, $\phi(t)$ and Gaussian noise $\epsilon(t)$

$$f(t, x(t)) = Lx(t) - \phi(t) + \epsilon(t)$$

we have

$$y(t) \equiv \dot{x}(t) - Lx(t) - \epsilon(t) = \phi(t), \qquad \text{where} \quad \dot{x}(t) \equiv \frac{d}{dt}x(t) \tag{3}$$

Since $x(t)$ is assumed a GP, and $y(t)$ is a linear function of $x(t)$ and $\epsilon(t)$, then $y(t)$ is also a GP. The covariance function of this new process is straightforward to obtain using the standard rules, see [7]. For example, the covariance terms involving $\dot{x}$ are simply obtained by differentiating the covariance function of x:

$$\langle \dot{x}(t)x(t')\rangle_{p(\dot{x},x)} = \frac{\partial}{\partial t}C(t, t'), \qquad \langle \dot{x}(t)\dot{x}(t')\rangle_{p(\dot{x})} = \frac{\partial^2}{\partial t \partial t'}C(t, t') \tag{4}$$

where $\langle f(x)\rangle_{p(x)}$ denotes expectation of the function $f(x)$ with respect to the distribution $p(x)$. Given then observations $y_n \equiv \phi(t_n)$ at the given observation times and any boundary or initial conditions on x and $\dot{x}$, then $x(t)$ is a GP whose mean and covariance function is given by the standard Gaussian conditioning formulae

$$p(x|y) = \mathcal{N}\left(x\middle|\mu_x + C_{xy}C_{yy}^{-1}(y - \mu_y), C_{xx} - C_{xy}C_{yy}^{-1}C_{yx}\right) \tag{5}$$

In this way we can globally approximately solve the IVP (or BVP), giving a Gaussian distribution over the solution approximation $p(x_{2:N}|x_1, \mathcal{D})$. Whilst this method has cubic complexity in the number of points N that we need to evaluate the function at, this will typically be much smaller than the number of timepoints used in a standard ODE solver, provided that the solution is sufficiently smooth.

1.2 Skilling's IVP approach for non-linear ODEs

In the case that f is not linear in x, the problem is generally much more complex. One approach would be to assume that the approximation follows a GP and perform local linearisation, analogous to Exponential Integrators, see for example [6]. However, recent work in this area [5, 2] has developed the suggestion by Skilling [9], which we outline below.

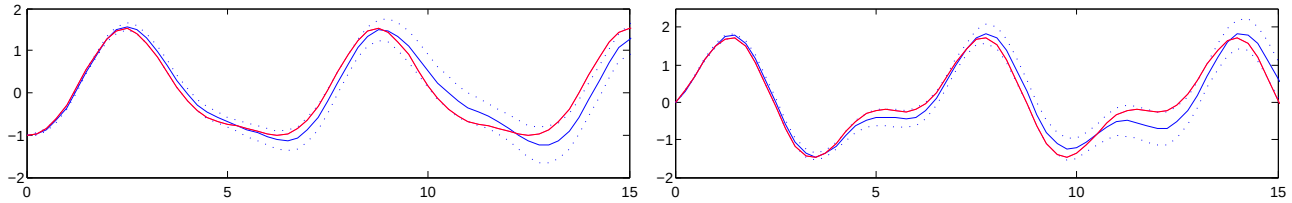
The fundamental quantity of interest in Skilling's [9] approach[3] for the IVP is the set of derivatives $\dot{\mathbf{x}} \equiv \dot{x}_1, \ldots, \dot{x}_N$ at specified 'knotpoints' $t_1, \ldots, t_N$. In [5, 2] a GP is assumed for the approximate solution $x_{1:N}$. We start with the known initial state x_1 and compute its derivative[4] $\dot{x}_1 = f(x_1)$. This is the only point and derivative that we know with certainty. One can interpret this as an observation of the derivative with zero observation error, $\sigma_1^2 = 0$. We assume a zero mean GP with known covariance function[5]. Using the GP we can form a distribution for the solution at the next knotpoint

$$p_{GP}(x_2|x_1, \dot{x}_1) \tag{6}$$

We now sample a value for x_2 from (6) and subsequently compute the derivative $\dot{x}_2 = f(x_2)$. Note that both the value x_2 and derivative $\dot{x}_2$ will not necessarily correspond to the true solution and its derivative. Because of this, only the derivative is retained and the interpretation is that one has observed the derivative $\dot{x}_2$ with measurement error σ_2^2 specified by the variance of $p_{GP}(\dot{x}_2|x_1, \dot{x}_1)$. Given $x_1, \dot{x}_1 = f(x_1), \dot{x}_2 = f(x_2)$ and the corresponding variances on these observations $\sigma_{1:2}^2$, one now forms the GP prediction

$$p_{GP}(x_3|, x_1, \dot{x}_1 = f(x_1), \dot{x}_2 = f(x_2), \sigma_{1:2}^2) \tag{7}$$

As before one then samples a value for x_3 and subsequently computes the derivative observation $\dot{x}_3 = f(x_3)$ which is assumed to be measured with observation variance obtained from $p_{GP}(\dot{x}_3|\dot{x}_{1:2}, \sigma_{1:2}^2)$. One continues



(a) $x_1(t)$ (b) $x_2(t)$

Figure 1: Solutions to the ODE (8) for $\theta = 2$. Plotted are the solution (left) and its derivative (right). The exact solution is plotted in red. Two solution methods are shown: Runge-Kutta4.5 (magenta), which is virtually indistinguishable from the exact solution, and the Skilling GP approach with stepsize $\delta = 0.25$. Plotted in dashed lines are the estimated one standard deviation errors in the GP solution.

in this manner defining a set of derivative observations $\dot{x}_{1:N}$ and corresponding observation noises $\sigma^2_{1:N}$. These can then be used as part of a standard GP prediction model to infer $p_{GP}(x_{2:N}|x_1, \dot{x}_{1:n}, \sigma^2_{1:n})$.

Whilst this procedure can be shown to retrieve the exact integrated curve in the limit of infinitely densely spaced knotpoints [2], the naive time complexity is $O\left(N^3\right)$ (due to Gaussian conditioning) which would most likely make this much slower than standard ODE solvers. This complexity can however be reduced by using more specialised covariance functions, see [2]. In figure(1) we show this approach applied to solving the ODE[6]

$$f_1(t) = x_2(t), \qquad f_2(t) = -x_1(t) + \sin(\theta t); \tag{8}$$

which has exact solution

$$x_1(t) = \left(-\theta^2 \cos(t) + \theta \sin(t) - \sin(\theta t) + \cos(t)\right)/(\theta^2 - 1), \quad x_2(t) = dx_1(t)/dt$$

For this experiment (and throughout the paper) we used the squared exponential covariance $C(t, t') = \exp(-(t - t')^2)$. From figure(1) we see that the Skilling GP procedure is substantially worse in terms of numerical accuracy than the standard Runge Kutta approach. One potential reason for this is that it discards useful information gathered about the function, namely the sample values $x_2, \ldots, x_N$. These could also be included as 'noisy' measurements of the true integrated curve, with measurement error similarly given by the variance of the predicted GP. However, our experience is that extending the scheme in this manner does not significantly improve the accuracy of the approach. Given the drawbacks of this GP solution technique, we were motivated to consider alternative approaches for probabilistic solutions and uncertainty estimates in non-linear ODEs.

2 Novel ODE solvers

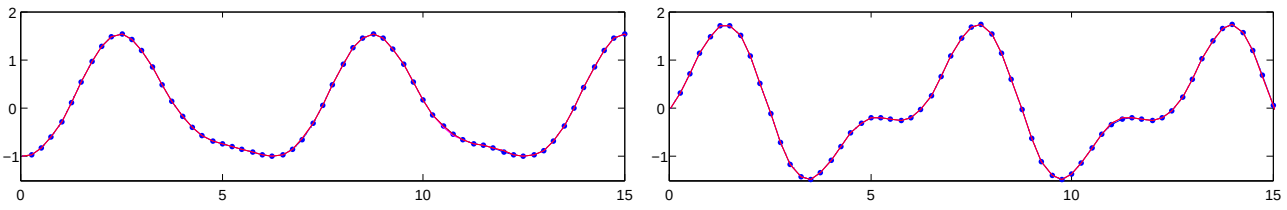
There are a great many directions that one could take in constructing a probabilistic solver and we outline only three. We also describe in section(2.5) a general method that can be used to estimate the error in any ODE solution (obtained from a standard ODE solver).

[3]It is perhaps worth mentioning that Skilling viewed his approach as only a suggestion amongst other potential related approaches.

[4]We drop the potential dependence of f on t to avoid notational clutter.

[5]The approach in [2] is slightly different – they do not condition on knowing $x_1, \dot{x}_1$ (see their equation 11). Rather they impose the mean of the GP at timestep 1 to be $x_1, \dot{x}_1$ with zero covariance. Subsequent timesteps have zero mean. We take the approach as outlined in [5] – there is little practical difference in the two approaches.

[6]Note that here the subscript denotes the component of the two dimensional vector exact solution.



(a) $x_1(t)$ (b) $x_2(t)$

Figure 2: Solutions to the ODE (8) for $\theta = 2$. Plotted are the solution (left) and its derivative (right). The exact solution is plotted in red and the Explicit GP approach with stepsize $\delta = 0.25$ is plotted in blue dots (indistinguishable from the the exact solution). The error estimates are too small to be visible.

2.1 An Explicit Solver

Our first solver follows most closely in spirit to Skilling's approach. Given $x_1, \dot{x}_1$, we would like to find $p(x_2|x_1, \dot{x}_1)$. We can obtain this using[7]

$$\begin{aligned} p(x_2|x_1,\dot{x}_1) &= \int_{\dot{x}_2} p(x_2,\dot{x}_2|x_1,\dot{x}_1) = \int_{\dot{x}_2} p_{GP}(x_2|\dot{x}_2,x_1,\dot{x}_1)p(\dot{x}_2|x_1,\dot{x}_1) \\ &= \int_{\dot{x}_2} p_{GP}(x_2|\dot{x}_2,x_1,\dot{x}_1)\int_{\tilde{x}_2} p(\dot{x}_2,\tilde{x}_2|x_1,\dot{x}_1) \\ &= \int_{\dot{x}_2} p_{GP}(x_2|\dot{x}_2,x_1,\dot{x}_1)\int_{\tilde{x}_2} p_{ODE}(\dot{x}_2|\tilde{x}_2,x_1,\dot{x}_1)p_{GP}(\tilde{x}_2|x_1,\dot{x}_1) \\ &= \int_{\dot{x}_2,\tilde{x}_2} p_{GP}(x_2|\dot{x}_2,x_1,\dot{x}_1)p_{ODE}(\dot{x}_2|\tilde{x}_2,x_1,\dot{x}_1)p_{GP}(\tilde{x}_2|x_1,\dot{x}_1) \end{aligned}$$

For simplicity we assume a deterministic ODE so that we can write

$$\begin{aligned} p(x_2|x_1,\dot{x}_1) &= \int_{\dot{x}_2,\tilde{x}_2} p_{GP}(x_2|\dot{x}_2,x_1,\dot{x}_1)\delta(\dot{x}_2 - f(\tilde{x}_2))p_{GP}(\tilde{x}_2|x_1,\dot{x}_1) \\ &= \int_{\tilde{x}_2} p_{GP}(x_2|\dot{x}_2 = f(\tilde{x}_2),x_1,\dot{x}_1)p_{GP}(\tilde{x}_2|x_1,\dot{x}_1) \end{aligned}$$

We can then obtain samples from $p(x_2|x_1, \dot{x}_1)$ by forward sampling: we sample a putative future value of the solution using the GP $p_{GP}(\tilde{x}_2|x_1, \dot{x}_1)$, conditioned on past information. However, this value does not necessarily satisfy the derivative requirement of the ODE. To see how well it matches, we calculate what the derivative $\dot{x}_2$ of this putative solution should be. Given this we can then sample a value for x_2. In this manner the generated x_2 will be consistent with the smoothness assumption of the GP and also consistent with the derivative requirement of the ODE[8].

More generally, given a sample from the distribution

$$p(x_n|x_{1:n-1},\dot{x}_1)$$

the distribution $p(x_{n+1}|x_{1:n}, \dot{x}_1)$ is recursively defined by, see algorithm(1),

$$\begin{aligned} &p(x_{n+1}|x_{1:n},\dot{x}_1) \\ &\qquad = \int_{\tilde{x}_{n+1}} p_{GP}(x_{n+1}|x_{1:n},\dot{x}_{1:n} = f(x_{1:n}),\dot{x}_{n+1} = f(\tilde{x}_{n+1}))p_{GP}(\tilde{x}_{n+1}|x_{1:n},\dot{x}_{1:n} = f(x_{1:n})) \end{aligned}$$

Given $x_{1:n}$ and $\dot{x}_{1:n}$, we then sample a state $\tilde{x}_{n+1}$ from $p_{GP}(\tilde{x}_{n+1}|x_{1:n}, \dot{x}_{1:n})$ and subsequently a state x_{n+1} from $p_{GP}(x_{n+1}|, x_{1:n}, \dot{x}_{1:n} = f(x_{1:n}), \dot{x}_{n+1} = f(\tilde{x}_{n+1}))$. We repeat this process until time index N, which defines then a single trajectory $x_{2:N}$. To define another solution sample $x_{2:N}$, we repeat the above process beginning from time index $n = 1$. The distribution over solutions is then formally obtained by $\prod_n p(x_n|x_{1:n-1}, \dot{x}_1)$.

[7] We write out the steps explicitly to explain the intuition behind the derivation.

[8] Although we do not do so here, one can consider variations on this theme such as including additional putative values such as $\tilde{\tilde{x}}_2$ *etc.*

Algorithm 1 Explicit Multistep ODE solver . Draw S samples, each with an M length history.

for $l = 1 : S$ **do** ▷ Sample multiple trajectories
  for $n = 1 : N$ **do** ▷ Sample a single trajectory
    Draw a sample $\tilde{x}_{n+1}$ from $p_{GP}(\tilde{x}_{n+1}|x_{n-M:n}, \dot{x}_{n-M:n})$
    Compute the derivative at this point $\dot{x}_{n+1} = f(\tilde{x}_{n+1}))$
    Draw x_{n+1} from $p_{GP}(x_{n+1}|, x_{n-M:n}, \dot{x}_{n-M:n} = f(x_{n-M:n}), \dot{x}_{n+1} = f(\tilde{x}_{n+1}))$.
  end for
  This defines a sample $x^l_{1:N}$
end for

Algorithm 2 Implicit Multistep ODE solver . Draw a sample solution

Initialise the sample $x^1_{2:N}$
for $i = 1 : I$ **do** ▷ Iteration counter
  for $n = 2 : N$ **do** ▷ Sample a trajectory
    Compute the derivative at each point $\dot{x}_n = f(x^i_n)$
  end for
  Draw a sample $x_{2:N}$ from $p_{GP}(x_{2:N}|\dot{x}_{1:n}, x_1)$.
end for
After I iterations we have a sample $x_{1:N}$

Note that this procedure differs significantly from Skilling's. Firstly, points are generated that are more likely to be consistent with the ODE requirement. Also, by conditioning on the past samples, we can limit the time horizon for the GP prediction. That is, at timestep n, rather than conditioning on all past observations $x_{1:n-1}$ (which would have computational complexity cubic in n) we can limit the conditioning to say M previous observations, limiting the complexity of drawing a sample for x_n to cubic complexity in M. This is a significant improvement in complexity than previous approaches and brings the method in line with standard multistep ODE solver complexities.

We demonstrate the method in figure(2) which has the same setup as Skilling's approach in figure(1). Despite using the same stepsize $\delta = 0.25$ in both approaches, the explicit GP method has excellent comparative performance and is computationally significantly cheaper.

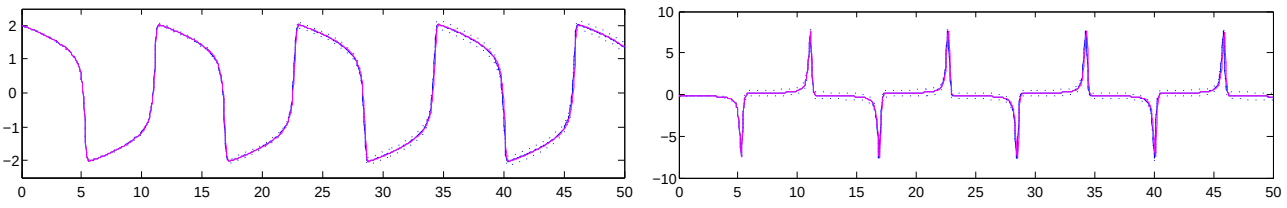
2.2 An Implicit Solver

A drawback of explicit approaches is that they can lack consistency and also stability [8]. One way to view deriving consistent approximations is to require that if we solve going forwards, and then using this solution reverse time and solve backwards, we should end up where we started from[9]. A related approach is to assume a solution that must be globally consistent[10]. We will first assume that we wish to sample a trajectory $p(x_2, x_3|x_1, \dot{x}_1)$. We can write (for a deterministic ODE)

$$\begin{aligned}
p(x_2, x_3|x_1, \dot{x}_1) &= \int_{\dot{x}_2, \dot{x}_3} p(x_2, x_3, \dot{x}_2, \dot{x}_3|x_1, \dot{x}_1) \\
&= \int_{\dot{x}_2, \dot{x}_3} p(x_2, x_3|x_1, \dot{x}_{1:3}, x_1) p(\dot{x}_2, \dot{x}_3|x_1, \dot{x}_1) \\
&= \int_{\tilde{x}_2, \tilde{x}_3, \dot{x}_2, \dot{x}_3} p_{GP}(x_2, x_3|x_1, \dot{x}_{1:3}) p_{ODE}(\dot{x}_2, \dot{x}_3|\tilde{x}_2, \tilde{x}_3, x_1, \dot{x}_1) p(\tilde{x}_2, \tilde{x}_3|x_1, \dot{x}_1) \\
&= \int_{\tilde{x}_2, \tilde{x}_3} p_{GP}(x_2, x_3|\dot{x}_1, \dot{x}_2 = f(\tilde{x}_2), \dot{x}_3 = f(\tilde{x}_3), x_1) p(\tilde{x}_2, \tilde{x}_3|x_1, \dot{x}_1)
\end{aligned}$$

Thus, if we start with a distribution $p(\tilde{x}_2, \tilde{x}_3|x_1, \dot{x}_1)$ over solutions, the above updates this to a new distribution $p(x_2, x_3|x_1, \dot{x}_1)$. This is analogous to Picard iteration, see for example [8], and can be perhaps best considered as a distributional approximation to the Picard approach. By recursing, we seek the fixed point solution $p^*(x_2, x_3|x_1, \dot{x}_1)$ of the above procedure. The fixed point then has the required global consistency property[11].

[9] This is the intuition behind for example the mid-point extension of the Euler method.
[10] This is essentially the approach taken by Picard iteration.
[11] Intuitively, in the limit of small δ this tends to the Picard iteration and thus to the exact solution, due to the contraction property of the Picard operator.



(a) $x_1(t)$ (b) $x_2(t)$

Figure 3: Van der Pol oscillator (9) with $\theta = 5$. Plotted are the solution (left) and its derivative (right). Two solution methods are shown: Runge-Kutta4.5 (magenta) and the Implicit GP approach with window length 5 and stepsize $\delta = 0.05$. Plotted in dashed lines are the estimated errors in the GP solution.

Since the above updating process is not closed with respect to any standard distribution class, one could alternatively draw samples recursively, see algorithm(2). Another approach, which we adopt in the experiments, is to assume that $p(\tilde{x}_2, \tilde{x}_3|x_1, \dot{x}_1)$ is approximated by a Gaussian with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. We then evaluate

$$p_{GP}(x_2, x_3|\dot{x}_1, \dot{x}_2 = f(\mu_2), \dot{x}_3 = f(\mu_3), x_1)$$

which defines the new mean and covariance. We iterate this to convergence.

In principle, one can apply the above to the whole solution trajectory. However, this is computationally expensive, scaling $O\left(N^3\right)$ due to the Gaussian conditioning step. An alternative is to consider a small size M window and solve for the future values in this small window. Then the window is moved forward one timestep. For our example above, this would define a distribution for x_2 and x_3. We could then move forward one timestep with x_2 replacing x_1 as the conditioned information. Similarly, rather than x_1 being observed with certainty, we assume that x_2 is observed with variance obtained from $p^*(x_2|x_1, \dot{x}_1)$. As we move forwards, we can retain a limited history of the computed values to bracket the variable x_n by a small number of past and future variables, analogous to implicit multistep solvers [8].

As an example we show in figure(3) the solution technique applied to the Van der Pol oscillator.

$$f_1(x(t)) = x_2(t), \qquad f_2(x(t)) = -x_1(t) + \theta\left(1 - x_1^2(t)\right) x_2(t) \tag{9}$$

As we can see, the approach performs well, with increasing uncertainty as time increases. Compared to the Explicit GP approach (not shown) the Implicit approach solves this problem more accurately, though with a larger number of function evaluations due to the fixed point iteration.

2.3 Implicit Gradient Matching

If we assume we are given a proposed solution $x_{1:N}$, we can use a GP to calculate the derivative distribution of this point sample curve,

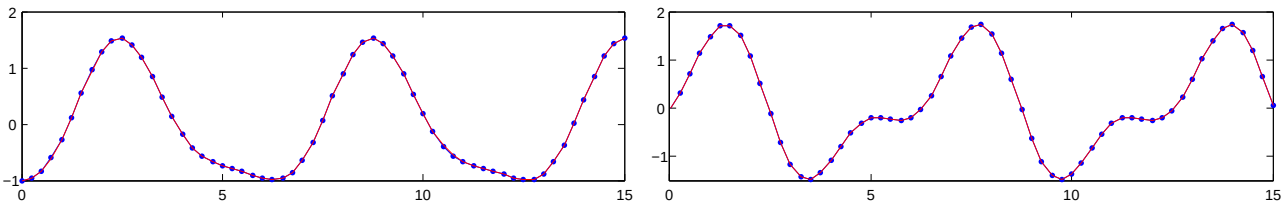
$$p_{GP}(\dot{x}_{2:N}|\dot{x}_1, x_{1:N})$$

A self consistency requirement is that these derivatives should match the known ODE derivatives $f(x_n)$ at the points $t = 2, \ldots, N$. The expected mismatch is

$$E(x_{2:T}) \equiv \left\langle \sum_{\tau=2}^{N} \left(f(x_\tau) - \dot{x}_\tau\right)^2 \right\rangle_{p_{GP}(\dot{x}_{2:N}|\dot{x}_1, x_{1:N})} = \sum_{\tau=2}^{N} \left(f(x_\tau) - \langle \dot{x}_\tau|\dot{x}_1, x_{1:N}\rangle\right)^2 + \sigma^2(\dot{x}_\tau)$$

The term $\langle \dot{x}_\tau|\dot{x}_1, x_{1:N}\rangle$ denotes the mean of the variable $\dot{x}_\tau$ conditioned on knowing $\dot{x}_1, x_{1:N}$. For a GP, the final variance term $\sigma^2(\dot{x}_\tau)$ is independent of $x_{2:N}$. Also for a GP, the predicted mean is a linear function of the observation and thus the mean of $\dot{x}_\tau$ is a linear function of $\dot{x}_1, x_{1:n}$. An equivalent optimisation problem is to minimise with respect to $\mathbf{x}$

$$F(\mathbf{x}) \equiv \sum_{\tau=2}^{N} \left(f(x_\tau) - c_\tau - \mathbf{a}_\tau^\mathsf{T}\mathbf{x}\right)^2$$



(a) $x_1(t)$ (b) $x_2(t)$

Figure 4: Solutions to the ODE (8) for $\theta = 2$. Plotted are the solution (left) and its derivative (right). The exact solution is plotted in red. The gradient matching approach with window of length 10 is shown with stepsize $\delta = 0.25$ and is virtually indistinguishable from the exact solution.

where $\mathbf{x} = x_{2:N}$ for suitably defined vectors $\mathbf{a}_\tau$ and constants c_τ (these are simply derived from the GP). The optimisation can be achieved by standard approaches. In our experiments, we formed an update based on equating the derivative of F to zero; this gives a rapidly converging estimate. Whilst one can in principle carry out this optimisation for all $x_{2:N}$, this is wasteful since only timepoints close to the initial time will be relevant for determining the solution close to the initial time (the solution method will first determine x_2 and then x_3, *etc.*). For this reason we therefore considered a windowed approach, moving the solution forward by one timestep once a convergence criterion for the window is passed. An example is given in figure(4). In our experience, this gradient matching approach performs well, but is less accurate than the implicit GP approach.

2.4 Using Higher Order Derivative Information

One benefit of the GP approach is that it is straightforward to extend to conditioning on higher order derivatives. Given the collection of ODEs,

$$\frac{d}{dt}x_i = f_i(t, x)$$

we can compute

$$\frac{d^2}{dt^2}x_i = \frac{\partial}{\partial t}f_i(t,x) + \sum_j \frac{d}{dx_j}f_i(t,x)\frac{d}{dt}x_j = \frac{\partial}{\partial t}f_i(t,x) + \sum_j J_{ij}(t,x)f_j(t,x)$$

where the Jacobian is defined

$$J_{ij}(t,x) \equiv \frac{d}{dx_j}f_i(t,x)$$

We can then use these second order derivatives $\ddot{x}$ as part of the GP conditioning set. Our code includes the option of using higher order derivative information in any of the above three novel solvers.

2.5 Deriving Error Estimates

Given an approximate solution $x_{2:N}$ for the IVP from a standard ODE solver, we can estimate the error as follows. We first compute $p_{GP}(\dot{x}_{2:N}|\dot{x}_1, x_{1:N})$ and draw a sample $\dot{x}_{2:N}$ from this. If our solution $x_{2:N}$ were correct, then the derivative should be $f(x_{2:N})$. We can therefore obtain a local estimate of the error in the derivative $\dot{x}_n$ by

$$\dot{\sigma}_n^2 \equiv \left\langle (f(x_n) - \dot{x}_n)^2 \right\rangle_{p_{GP}(\dot{x}_n|\dot{x}_1, x_{1:N})} \tag{10}$$

Using this we can then form the Gaussian likelihood for an ODE solution $p_{GP}(x_{2:N}|x_1, \dot{x}_1, \dot{x}_{2:N} = f(x_{2:N}))$ in which during the GP conditioning it is assumed that the derivatives are observed with the variances computed by (10). This likelihood can be used to assess the quality of the ODE solution $x_{2:N}$.

3 Discussion and Summary

There has been recent interest in approximate methods for solving ODEs based on using Gaussian Processes. We have noted that the approaches [2, 5] based on Skilling's suggestion [9] suffer some drawbacks and it is unclear if they can be made practically useful in their current form. In contrast, we suggested a collection of techniques based on insights from standard ODE solvers, using both implicit and explicit information. To date we have carried out only limited experiments but believe these are promising directions to consider as alternatives to existing GP approaches for solving ODEs.

The simplest of our approaches is the Explicit GP method which has reasonable accuracy and is analogous to Explicit multistep ODE solvers. In our experience, as would be expected from such a simple forward explicit approach, the accuracy is lower (for a similar number of function evaluations) than can be achieved by more sophisticated implicit techniques.

Our Implicit GP method is also straightforward to implement, though is slightly more complex than the forward approach. However, the numerical accuracy of the approach is high. In our experiments on the Van der Pol oscillator, the method outperforms the Explicit approach and has accuracy similar to standard ODE solvers such as Runge Kutta 4.5.

The gradient matching approach is another implicit approach that also improves on the Explicit GP approach. In our experiments, we have found that the method has comparable accuracy to the Implicit approach though solving the required optimisation problem at each timestep is more costly than the fixed-point iteration of the Implicit GP approach.

The extension of these methods to solving partial differential equations, as in [2], is in principle straightforward.

Acknowledgements

I would like to thank Mark Girolami for helpful discussions.

References

[1] B. Calderhead, M. Girolami, and N. D. Lawrence. Accelerating Bayesian Inference over Nonlinear Differential Equations with Gaussian Processes. In *NIPS*, 2008.

[2] O. A. Chkrebtii, D. A. Campbell, M. A. Girolami, and B. Calderhead. Bayesian Uncertainty Quantification for Differential Equations. *ArXiv e-prints*, June 2013.

[3] F. Dondelinger, M. Filippone, S. Rogers, and D. Husmeier. ODE parameter inference using adaptive gradient matching with Gaussian processes. In *AISTATS*, 2013.

[4] T. Graepel. Solving noisy linear operator equations by Gaussian processes: application to ordinary and partial differential equations. In *ICML*, 2003.

[5] P. Hennig and S. Hauberg. Probabilistic Solutions to Differential Equations and their Application to Riemannian Statistics. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 33 of *JMLR: Workshop and Conference Proceedings*, 2014.

[6] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, (19):209–286, May 2010.

[7] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[8] J. C. Robinson. *An Introduction to Ordinary Differential Equations,*. Cambridge University Press, 2004.

[9] J. Skilling. Bayesian solution of ordinary differential equations. In C. R. Smith, G. J. Erickson, and P. O. Neudorfer, editors, *Maximum Entropy and Bayesian Methods*, pages 23–37, Dordrecht, 1991. Kluwer.

[10] Y. Wang and D. Barber. Gaussian Processes for Bayesian Estimation in Ordinary Differential Equations. In *Proceedings of the 31st International Conference on Machine Learning*, Beijing, China, 2014. JMLR: W&CP volume 32.