Computational Inference in Systems Biology

Benn Macdonald and Dirk Husmeier

University of Glasgow, Department of Mathematics and Statistics,
Scotland, G12 8QW
b.macdonald.1@research.gla.ac.uk,
Dirk.Husmeier@glasgow.ac.uk

Abstract. Parameter inference in mathematical models of biological pathways, expressed as coupled ordinary differential equations (ODEs), is a challenging problem. The computational costs associated with repeatedly solving the ODEs are often high. Aimed at reducing this cost, new concepts using gradient matching have been proposed. This paper combines current adaptive gradient matching approaches, using Gaussian processes, with a parallel tempering scheme, and conducts a comparative evaluation with current methods used for parameter inference in ODEs.

Keywords: Parameter Inference, Ordinary Differential Equations, Gradient Matching, Parallel Tempering, Gaussian Processes.

1 Introduction

Ordinary differential equations (ODEs) have many applications in systems biology. Conventional inference methods involve numerically integrating the system of ODEs, to calculate the likelihood as part of an iterative optimisation or sampling procedure. However, the computational costs involved with numerically solving the ODEs are large. To reduce the computational complexity, several authors have adopted an approach based on gradient matching (e.g. [1] and [15]). The idea is based on the following two-step procedure. In a first, preliminary smoothing step, the time series data are interpolated; in a second step, the kinetic parameters $\boldsymbol{\theta}$ of the ODEs are optimised so as to minimise some metric measuring the difference between the slopes of the tangents to the interpolants, and the $\boldsymbol{\theta}$-dependent time derivative from the ODEs. In this way, the ODEs never have to be solved explicitly, and the typically unknown initial conditions are effectively profiled over. A disadvantage of this two-step scheme is that the results of parameter inference critically hinge on the quality of the initial interpolant. A better approach, first suggested in [13], is to regularise the interpolants by the ODEs themselves. Dondelinger et al. [4] applied this idea to the nonparametric Bayesian approach of [1], using Gaussian Processes (GPs), and demonstrated that it substantially improves the accuracy of parameter inference and robustness with respect to noise. As opposed to [13], all smoothness hyperparameters are consistently inferred in the framework of nonparametric Bayesian statistics, dispensing with the need to adopt heuristics and approximations.

F. Ortuño and I. Rojas (Eds.): IWBBIO 2015, Part II, LNCS 9044, pp. 276–288, 2015.

We extend the work of [4] in two respects. Firstly, we combine adaptive gradient matching using GPs with a parallel tempering scheme for the gradient mismatch parameter. This is conceptually different from the inference paradigm of the mismatch parameter that [4] employs. If the ODEs provide the correct mathematical description of the system, ideally there should be no difference between the interpolant gradients and those predicted from the ODEs. In practise, however, forcing the gradients to be equal is likely to cause parameter inference techniques to converge to a local optimum of the likelihood. A parallel tempering scheme is the natural way to deal with such local optima, as opposed to inferring the degree of mismatch, since different tempering levels correspond to different strengths of penalising the mismatch between the gradients. Since our modelling process is created using a products of experts approach (see section 2), parallel tempering should work well on penalising the mismatch. A parallel tempering scheme was explored by Campbell & Steele [3], however, their approach uses a different methodological paradigm, and thus the results are not directly comparable to the GP approach in [4]. In this paper, we present for the first time, a comparative evaluation of parallel tempering versus inference in the context of gradient matching for the same modelling framework, i.e. without any confounding influence from the model choice. Secondly, we compare the method of Bayesian inference with Gaussian Processes with a variety of other methodological paradigms, within the specific context of adaptive gradient matching, which is highly relevant to current computational systems biology.

2 Methodology

Consider a set of T arbitrary time points $t_1 < ... < t_T$, and a set of noisy observations $\mathbf{Y} = (\mathbf{y}(t_1), ..., \mathbf{y}(t_T))$, where $\mathbf{y}(t) = \mathbf{x}(t) + \boldsymbol{\epsilon}(t) + \boldsymbol{\mu}$, N = dim($\mathbf{x}(t)$), $\mathbf{X} = (\mathbf{x}(t_1), ..., \mathbf{x}(t_T))$. The signals of the system are described by ordinary differential equations (ODEs), of the form

$$\mathbf{x}' = \frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t) + \boldsymbol{\mu}, \boldsymbol{\theta}, t); \qquad \mathbf{x}(t_1) = \mathbf{x}_1 \tag{1}$$

where $\boldsymbol{\theta}$ is a parameter vector of length r, $\boldsymbol{\mu}$ is a vector of integration constants, for simplicity set as the sample mean, and $\boldsymbol{\epsilon}$ is multivariate Gaussian noise, $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma_n^2 \mathbf{I})$. Then,

$$p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\sigma}) = \prod_n \prod_t N(y_n(t)|x_n(t) + \mu_n, \sigma_n) \tag{2}$$

and the matrices $\mathbf{X}$ and $\mathbf{Y}$ are of dimension N by T. Now let $\mathbf{x}_n$ and $\mathbf{y}_n$ be T-dimensional column vectors containing the n^{th} row of $\mathbf{X}$ and $\mathbf{Y}$. Following [1], we place a Gaussian process (GP) prior on $\mathbf{x}_n$,

$$p(\mathbf{x}_n|\boldsymbol{\phi}) = N(\mathbf{x}_n|\mathbf{0}, \mathbf{C}_{\phi_n}) \tag{3}$$

where $\mathbf{C}_{\phi_n}$ is a positive definite matrix of covariance functions with hyperparameters ϕ_n. Assuming additive Gaussian noise, with state-specific error variance σ_n^2, we get $p(\mathbf{y}_n|\mathbf{x}_n, \sigma_n) = N(\mathbf{y}_n|\mathbf{x}_n, \sigma_n^2 \mathbf{I})$, and

$$p(\mathbf{y}_n|\boldsymbol{\phi}_n, \sigma_n) = \int p(\mathbf{y}_n|\mathbf{x}_n, \sigma_n)p(\mathbf{x}_n|\boldsymbol{\phi}_n)d\mathbf{x}_n = N(\mathbf{y}_n|\mathbf{0}, \mathbf{C}_{\phi_n} + \sigma_n^2\mathbf{I}) \quad (4)$$

The conditional distribution for the state derivatives is then

$$p(\mathbf{x_n}'|\mathbf{x_n}, \boldsymbol{\phi_n}) = N(\mathbf{m}_n, \mathbf{K}_n) \quad (5)$$

as the derivative of a GP is itself a GP, provided the kernel is differentiable [16], [1]. For closed form solutions to $\mathbf{m}_n$ and $\mathbf{K}_n$, see Rasmussen & Williams [10]. Assuming additive Gaussian noise with a state-specific error variance γ_n, from (1) we get

$$p(\mathbf{x}'_n|\mathbf{X}, \boldsymbol{\theta}, \gamma_n) = N(\mathbf{f}_n(\mathbf{X} + \boldsymbol{\mu}, \boldsymbol{\theta}), \gamma_n\mathbf{I}) \quad (6)$$

Dondelinger et al. [4] link the interpolant in (5) with the ODE model in (6) using a products of experts approach, obtaining a joint distribution

$$p(\mathbf{X}', \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) = p(\boldsymbol{\theta})p(\boldsymbol{\phi})p(\boldsymbol{\gamma}) \prod_n p(\mathbf{x}'_n|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \gamma_n)p(\mathbf{x}_n|\boldsymbol{\phi}_n) \quad (7)$$

They show that you can marginalise over the derivatives to get a closed form solution to

$$p(\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) = \int p(\mathbf{X}', \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma})d\mathbf{X}' \quad (8)$$

Using (2) and (8), our full joint distribution becomes

$$p(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\sigma}) = p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\sigma})p(\mathbf{X}|\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma})p(\boldsymbol{\theta})p(\boldsymbol{\phi})p(\boldsymbol{\gamma})p(\boldsymbol{\sigma}) \quad (9)$$

where Dondelinger et al. [4] show

$$\begin{aligned} p(\mathbf{X}|\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) = \frac{1}{\prod_n |2\pi(\mathbf{K}_n + \gamma_n\mathbf{I})|^{\frac{1}{2}}} \exp[-\frac{1}{2}\sum_n \\ (\mathbf{x}_n^T\mathbf{C}_{\phi_n}\mathbf{x}_n + (\mathbf{f}_n - \mathbf{m}_n)^T(\mathbf{K}_n + \gamma_n\mathbf{I})^{-1}(\mathbf{f}_n - \mathbf{m}_n))] \end{aligned} \quad (10)$$

$p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\sigma})$ is described in equation (2) and $p(\boldsymbol{\theta}), p(\boldsymbol{\phi}), p(\boldsymbol{\gamma}), p(\boldsymbol{\sigma})$ are the priors over the respective parameters. We use the same MCMC sampling scheme as [4], which uses the whitening approach of [20], to efficiently sample in the joint space of GP hyperparameters $\boldsymbol{\phi}$ and latent variables $\mathbf{X}$.

Parallel Tempering: The gradient matching approach is based on the intrinsic slack parameter γ (see equation (6)), which theoretically should be $\gamma = 0$, since this corresponds to no mismatch between the gradients. In practise, it is allowed to take on larger values, $\gamma > 0$, to prevent the inference scheme from getting stuck in sub-optimal states. However, rather than inferring γ like a model parameter, as carried out in [4], γ should be gradually set to $\gamma \to 0$. To this end we combine our gradient matching with Gaussian processes with the tempering approach in [3] (for details on tempering: [17], [18]) and temper this parameter to zero. Consider a series of "temperatures", $0 = \beta_1 < ... < \beta_M = 1$ and a power posterior distribution of our ODE parameters [14]

$$p_{\beta_i}(\boldsymbol{\theta}_i|\mathbf{y}) \propto p(\boldsymbol{\theta}_i)p(\mathbf{y}|\boldsymbol{\theta}_i)^{\beta_i} \quad (11)$$

(11) reduces to the prior for $\beta_i = 0$, and becomes the posterior when $\beta_i = 1$, with $0 < \beta_i < 1$ creating a distribution between our prior and posterior. The M approximations are used as the target densities of M parallel MCMC chains [3]. At each MCMC step, each chain independently performs a Metropolis-Hastings step to update $\boldsymbol{\theta}_i$. Also at each MCMC step, two chains are randomly selected, and a proposal to exchange parameters is made, with acceptance probability: $p_{swap} = min(1, \frac{p_{\beta_j}(\boldsymbol{\theta}_i|\mathbf{y})p_{\beta_i}(\boldsymbol{\theta}_j|\mathbf{y})}{p_{\beta_i}(\boldsymbol{\theta}_i|\mathbf{y})p_{\beta_j}(\boldsymbol{\theta}_j|\mathbf{y})})$. We now choose values of γ and fix them in place, associated with each β_i, for different chains, such that chains closer to the prior allow the gradients from the interpolant to have more freedom to deviate from those predicted by the ODEs, chains closer to the posterior to more closely match the gradients, and for β_M, we wish that the mismatch is approximately zero. Since γ corresponds to the variance of our state-specific error variance (see (6)), as $\gamma \to 0$, we have less mismatch between the gradients, and as γ gets larger, the gradients have more freedom to deviate from one another. Hence, we temper γ towards zero. Then each chain has a β_i for tempering of the power posterior and a γ_i for the gradient mismatch. For schedules, see table 3.

3 Alternative Methods for Comparison

We have carried out a comparative evaluation of the proposed scheme with various state-of-the-art gradient matching methods. These methods are based on different statistical modelling and inference paradigms: non-parametric Bayesian statistics with Gaussian processes without tempering, penalised regression splines, splines-based smooth functional tempering, and penalised likelihood based on reproducing kernel Hilbert spaces. Since many methods and settings are used in this paper for comparison purposes, for ease of reading, abbreviations are used. Table 1 is a reference for those methods and an overview of the methods is given below.

Table 1. Abbreviations of the methods used throughout this paper

Abbreviation	Method	Reference
C&S	Tempered mismatch parameter using splines-based smooth functional tempering (SFT)	Campbell & Steele [3]
GON	Reproducing kernel Hilbert space and penalised likelihood	González et al. [11]
INF	Inference of the gradient mismatch parameter using GPs	Dondelinger et al. [4]
LB2	Tempered mismatch parameter using GPs in Log Base 2 increments	Our method
LB10	Tempered mismatch parameter using GPs in Log Base 10 increments	Our method
RAM	Penalised splines & 2^{nd} derivative penalty	Ramsay et al. [13]

INF [4]: This method conducts parameter inference using adaptive gradient matching and Gaussian processes. The penalty mismatch parameter γ is inferred rather than tempered. **GON** [11]: Parameter inference is conducted in a non-Bayesian fashion, implementing a reproducing kernel Hilbert space (RKHS) and penalised likelihood approach. Comparisons between RKHS and GPs have been previously explored (for example, see [10], [19]) conceptually, and in this paper we analyse

Table 2. Particular settings of Campbell & Steele's [3] method

Abbreviation	Definition	Details
10C	10 Chains	When comparing our methods, it was of interest to see how the performance depended on the number or parallel MCMC chains, as originally the authors used 4 chains.
Obs20	20 Observations	Originally, the authors use 401 observations. We reduced this to a dataset size more usual with these types of experiments to observe the dependency of the methods on the amount of data.
15K	15 Knots	The method in C&S uses B-splines interpolation. We changed the original tuning parameters from the author's paper to observe the sensitivity of the parameter estimation by these tuning parameters.
P3	Polynomial order 3 (Cubic Spline)	The original polynomial order is 5 and again, we wanted to observe the sensitivity of the parameter estimation by these tuning parameters.

Table 3. Ranges of the penalty parameter γ for LB2 and LB10

Method	Chains	Range of Penalty γ	Method	Chains	Range of Penalty γ
LB2	4	[1 , 0.125]	LB10	4	[1 , 0.001]
LB2	10	[1 , 0.00195]	LB10	10	[1 , $1e^{-9}$]

this empirically in the specific context of gradient matching. The RKHS gradient matching method in [11] involves linearising the ODEs and obtaining a gradient using a differencing operator. **RAM** [13]: This method uses a non-Bayesian optimisation process for parameter inference. They use 2 penalties: the 2nd derivative of the interpolant (penalising by the curvature of the interpolant to avoid overfitting) and the difference between the gradients (using penalised splines). **C&S** [3]: Parameter inference is carried out using adaptive gradient matching and tempering of the mismatch parameter. The choice of interpolation scheme is B-splines. Table 2 outlines particular settings with some of the methods in table 1. The ranges of the penalty parameter for γ, for LB2 and LB10 methods are given in table 3. The increments are equidistant on the log scale. The M β_is from 0 to 1 are set, by taking a series of equidistant M values and raising them to the power 5 [14].

4 Data

Fitz-Hugh Nagumo ([5], [8])**:** These equations model the voltage potential across the cell membrane of the axon of giant squid neurons. There are two "species": Voltage (V) and Recovery variable (R), and 3 parameters; α, β and ψ. Species in [ ] denote the time-dependent concentration for that species:

$$\dot{V} = \psi([V] - \frac{[V]^3}{3} + [R]); \quad \dot{R} = -\frac{1}{\psi}([V] - \alpha + \beta * [R]) \tag{12}$$

Protein Signalling Transduction Pathway [12] **:** These equations model protein signalling transduction pathways [12] in a signal transduction cascade, where the free parameters are kinetic parameters governing how quickly the proteins

("species") convert to one another. There are 5 "species" (S, dS, R, RS, Rpp) and 6 parameters ($k_1, k_2, k_3, k_4, V, K_m$). The system describes the phosphorylation of a protein, $R \rightarrow Rpp$ (equation (17)), catalysed by an enzyme S, via an active protein complex (RS, equation (16)), where the enzyme is subject to degradation ($S \rightarrow dS$, equation (14)). The chemical kinetics are described by a combination of mass action kinetics (equations (13), (14) and (16)) and Michaelis-Menten kinetics (equations (15) and (17)). Species in [ ] denote the time-dependent concentration for that species:

$$\dot{S} = -k_1 * [S] - k_2 * [S] * [R] + k_3 * [RS] \tag{13}$$

$$\dot{dS} = k_1 * [S] \tag{14}$$

$$\dot{R} = -k_2 * [S] * [R] + k_3 * [RS] + \frac{V * [Rpp]}{K_m + [Rpp]} \tag{15}$$

$$\dot{RS} = -k_2 * [S] * [R] - k_3 * [RS] - k_4 * [RS] \tag{16}$$

$$\dot{Rpp} = k_4 * [RS] - \frac{V * [Rpp]}{K_m + [Rpp]} \tag{17}$$

5 Simulation

We have compared the proposed GP tempering scheme with the alternative methods summarised in Section 3. For those methods for which we were unable to obtain the software from the authors ([11] and [13]), we compared our results directly with the results from the original publications. To this end, we generated test data in the same way as described by the authors and used them for the evaluation of our method. For methods for which we did receive the authors' software ([3] and [4]), we repeated the evaluation twice, first on data equivalent to those used in the original publications, and again on new data generated with different (more realistic) parameter settings. For comparisons with other Bayesian methods, we used the authors' specifications for the priors on the ODE parameters. For comparisons with non-Bayesian methods, we applied our method with the parameter prior from [3], since the ODE model was the same. Our software is available upon request.

Reproducing Kernel Hilbert Space Method [11]: They tested their method on the Fitz-Hugh Nagumo data (see section 4) with the following parameters: $\alpha = 0.2; \beta = 0.2$ and $\psi = 3$. Starting from initial values of $(-1, -1)$ for the two "species", the authors generated 50 timepoints over the time course $[0, 20]$, producing 2 periods, with iid Gaussian noise (sd = 0.1) added. 50 independent data sets were generated in this way.

Penalised Splines & 2^{nd} Derivative Penalty Method [13]: This method was included in the study by [11], and we have used the results from their paper. For

comparison, our method was applied in the same way as for the comparison with [11].

Tempered Mismatch Parameter Using Splines-Based Smooth Functional Tempering [3]: They tested their method on the Fitz-Hugh Nagumo system with the following parameter settings: $\alpha = 0.2, \beta = 0.2$ and $\psi = 3$, starting from initial values of $(-1, 1)$ for the two "species". 401 observations were simulated over the time course [0, 20] (producing 2 periods) and Gaussian noise was added with sd $\{0.5, 0.4\}$ to each respective "species". In inferring the ODE parameters with their model, the authors chose the following settings: splines of polynomial order 5 with 301 knots; four parallel tempering chains associated with gradient mismatch parameters {10,100,1000,10000}; parameter prior distributions for the ODE parameters: $\alpha \sim N(0, 0.4^2)$, $\beta \sim N(0, 0.4^2)$ and $\psi \sim \chi^2_2$.

In addition to comparing our method with the results the authors had obtained with their settings, we made the following modifications to test the robustness of their procedure with respect to these (rather arbitrary) choices. We reduced the number of observations from 401 to 20 over the time course [0, 10] (producing 1 period) to reflect more closely the amount of data typically available from current systems biology projects. For these smaller data sets, we reduced the number of knots for the splines to 15 (keeping the same proportionality of knots to data points as before), and we tried a different polynomial order: 3 instead of 5. Due to the high computational costs of their method (roughly $1\frac{1}{2}$ weeks for a run), we could only repeat the MCMC simulations on 3 independent data sets. The respective posterior samples were combined, to approximately marginalise over data sets and thereby remove their potential particularities. For a fair comparison, we mimicked the authors' tempering scheme and only applied our method with 4 rather than 10 chains.

Inference of the Gradient Mismatch Parameter Using GPs and Adaptive Gradient Matching [4]: We applied the method in the same way as described in [4], using the authors' software and selecting the same kernels and parameter/hyperparameter priors as for the method proposed in the present paper. Data were generated from the protein signal transduction pathway described in Section 4. We applied our methods to data simulated from the same system, with the same settings as in [4]; ODE parameters: $(k_1 = 0.07, k_2 = 0.6, k_3 = 0.05, k_4 = 0.3, V = 0.017, K_m = 0.3)$; initial values of the species: $(S = 1, dS = 0, R = 1, RS = 0, Rpp = 0)$; 15 time points covering one period, $\{0, 1, 2, 4, 5, 7, 10, 15, 20, 30, 40, 50, 60, 80, 100\}$. Following [4], we added multiplicative iid Gaussian noise of standard deviation $\{0.1\}$ to all observations. For Bayesian inference, we chose the same gamma prior on the ODE parameters as used in [4], namely $\Gamma(4, 0.5)$. For the GP, we used the same kernel; see below for details. In addition to this ODE system, we also applied this method to the set-ups previously described for the Fitz-Hugh Nagumo model.

Choice of Kernel: For the GP, we need to choose a suitable kernel, which defines a prior distribution in function space. Two kernels were considered in our study (to match the authors' set-ups in [4]), the radial basis function (RBF) kernel

$$k(t_i, t_j) = \sigma^2_{RBF} exp(-\frac{(t_i - t_j)^2}{2l^2}) \tag{18}$$

with hyperparameters σ^2_{RBF} and l^2, and the sigmoid variance kernel

$$k(t_i, t_j) = \sigma^2_{sig} arcsin \frac{a + (bt_i t_j)}{\sqrt{(a + (bt_i t_i) + 1)(a + (bt_j t_j) + 1)}} \tag{19}$$

with hyperparameters σ^2_{sig}, a and b [10].

To choose initial values for the hyperparameters, we fit a standard GP regression model (i.e. without the ODE part) using maximum likelihood. We then inspect the interpolant to decide whether it adequately represents our prior knowledge. For the data generated from the Fitz-Hugh Nagumo model, we found that the RBF kernel provides a good fit to the data. For the protein signalling transduction pathway, we found that the non-stationary nature of the data is not represented properly with the RBF kernel, which is stationary [10], in confirmation of the findings in [4]. Following [4], we tried the sigmoid variance kernel, which is non-stationary [10] and we found this provided a considerably improved fit to the data. **Other settings:** Finally, the values for our variance mismatch parameter of the gradients, γ, needs to be configured. Log_2 and Log_{10} increments were used (with an initial start at 1), since studies that indicate reasonable values for our technique are limited (see [1], [14]). All parameters were initialised with a random draw from the respective priors (apart from GON, which did not use priors. For details of their technique, see [11]).

6 Results

Reproducing Kernel Hilbert Space [11] and **Penalised Splines & 2^{nd} Derivative Penalty Methods** [13]: For this configuration, to judge the performance of our methods, we used the same concept as in GON to examine our results. For each parameter, the absolute value of the difference between an estimator and the true parameter ($|\hat{\theta}_i - \theta_i|$) was computed and the distribution across the datasets were examined. For the LB2, LB10 and INF methods, the median of the sampled parameters was used as an estimator, since it is a robust estimator. Looking at figure 1 left, the LB2, LB10 and INF methods, do as well as the GON method, for 2 parameters (INF doing slightly worse for ψ) and outperform it for 1 parameter. All methods outperform the RAM method.

Tempered Mismatch Parameter Using Splines-Based Smooth Functional Tempering [3]: The C&S method shows good performance over all parameters in the one case where the number of observations is 401, the number of knots is 301 and the polynomial order is 3 (cubic spline), since the bulk of the distributions of the sampled parameters surround the true parameters in figures 1 right and 2 right and are close to the true parameter in figure 2 left. However, these settings require a great deal of "hand-tuning" or time expensive cross-validation and would be very difficult to set when using real data. The sensitivity of the splines based method can be seen in the other settings, where the results deteriorate. It

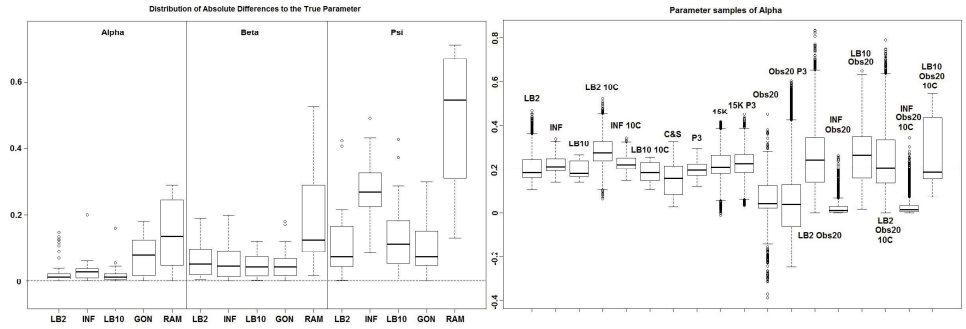

Fig. 1. Left: Boxplots of absolute differences to the true parameter over 50 datasets. The three sections from left to right represent the parameters α, β and ψ from equations (12). Within each section, the boxplots from left to right are: LB2 method, INF method, LB10 method, GON's method (boxplot reconstructed from [11]) and RAM's method (boxplot reconstructed from [11]). **Right:** Distributions of sampled Alphas from equation (12) over 3 datasets. From left to right: LB2, INF, LB10, LB2 10C, INF 10C, LB10 10C, C&S, C&S P3, C&S 15K, C&S 15K P3, C&S Obs20, C&S Obs20 P3, LB2 Obs20, INF Obs20, LB10 Obs20, LB2 Obs20 10C, INF Obs20 10C and LB10 Obs20 10C. The solid line is the true parameter. For definitions, see tables 1 and 2.

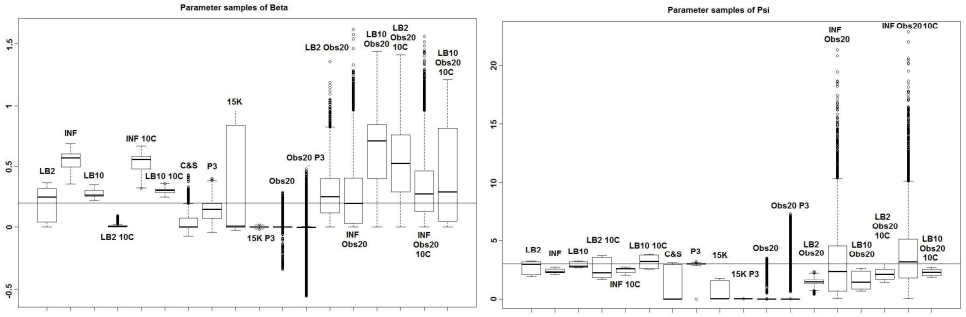

Fig. 2. Left: Distributions of sampled Betas from equation (12) over 3 datasets. From Left to right: LB2, INF, LB10, LB2 10C, INF 10C, LB10 10C, C&S, C&S P3, C&S 15K, C&S 15K P3, C&S Obs20, C&S Obs20 P3, LB2 Obs20, INF Obs20, LB10 Obs20, LB2 Obs20 10C, INF Obs20 10C and LB10 Obs20 10C. The solid line is the true parameter. **Right:** Distributions of sampled Psis from equations (12) over 3 datasets. From Left to right: LB2, INF, LB10, LB2 10C, INF 10C, LB10 10C, C&S, C&S P3, C&S 15K, C&S 15K P3, C&S Obs20, C&S Obs20 P3, LB2 Obs20, INF Obs20, LB10 Obs20, LB2 Obs20 10C, INF Obs20 10C and LB10 Obs20 10C. The solid line is the true parameter. For definitions, see tables 1 and 2.

is also important to note that when the dataset size was reduced, the cubic spline performed very badly. This inconsistency makes these methods very difficult to apply in practise. The LB2, LB10 and INF methods consistently outperform the C&S method with distributions overlapping or being closer to the true parameters. On the set-up with 20 observations, for 4 chains and 10 chains, the INF method produced largely different estimates over the datasets, as depicted by the wide boxplots

and long tails. The long tails in all these distributions are due to the combination of estimates from different datasets.

Inference of the Gradient Mismatch Parameter Using GPs, Adaptive Method [4]: In order to see how our tempering method performs in comparison to the INF method, we can examine the results from the protein signalling transduction pathway (see section 4), as well as comparing how each method did in the previous set-ups. The distributions of parameter estimates minus the true values for the protein signalling transduction pathway are shown in figure 3 left. The author's code was unable to converge properly for some of the datasets, so in order to present a clear indication of the methods' performance, we show the distributions across the dataset that showed the median parameter estimation, as determined by root mean square of the parameter samples.

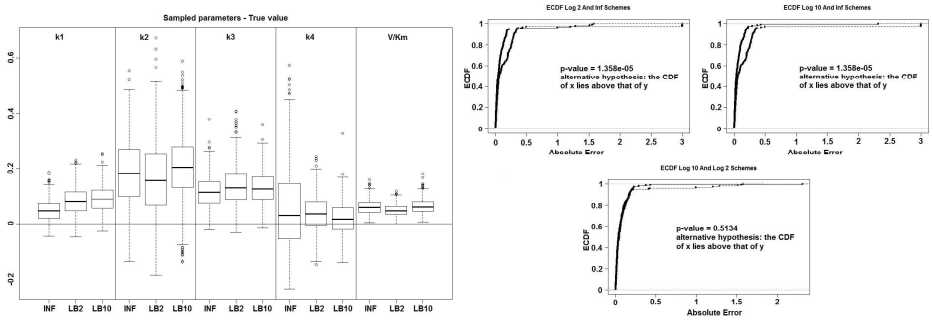

Fig. 3. Left: Average performance of the parameter inference for the INF, LB2 and LB10 methods. The distributions are of the sampled parameters from equations (13)-(17) minus the true value. The horizontal line shows zero difference. **Right:** ECDFs of the absolute errors of the parameter estimation. Top left - ECDFs for LB10 and INF, Top right - ECDFs for LB2 and INF and Bottom left - ECDFs for LB10 and LB2. Included are the p-values for 2-sample, 1-sided Kolmogorov-Smirnov tests. For definitions, see tables 1 and 2.

By examining figure 3 left, we can see that for each parameter, the distributions are close to the true values and so the methods are performing reasonably. Overall, there does not appear to be a significant difference between the INF, LB2 and LB10 methods for this model.

For the set-up in [11] and [13]: Figure 3 right shows the Expected Cumulative Distribution Functions (ECDFs) of the absolute errors of the parameter samples for the tempering and inference schemes. P-values for 2-sample, 1-sided Kolmogorov-Smirnov tests. If a distribution's ECDF is significantly higher than anothers, this constitutes better parameter estimation, since the distributions are of the average error. A higher curve means that more values are located in the lower range of absolute error.

Figure 3 right shows that both the LB2 and LB10 methods outperform the INF method, shown by p-values of less than the standard significance level of 0.05. Therefore we conclude that the CDFs for LB2 and LB10 are significantly higher than

those for INF. Since we are dealing with absolute error, this means that the parameter estimates from the LB2 and LB10 methods are closer to the true parameters than the INF method. The LB2 and LB10 method show no significant difference to each other.

For the set-up in [3]: The LB2 and LB10 methods do well over all the parameters and dataset sizes, with most of the mass of the distributions surrounding or being situated close to the true parameters. The LB2 does better than the LB10 for 4 parallel chains (distributions overlapping the true parameter for all three parameters) and the LB10 outperforms the LB2 for 10 parallel chains (distribution overlapping true parameter in figure 1 right, being closer to the true parameter in figure 2 left and narrower and more symmetric around the true parameter in figure 2 right). The INF method's bulk of parameter sample distributions are located close to the true parameters for all dataset sizes. However, the method produces less uncertainty at the expense of bias. When reducing the dataset to 20 observations, for 4 and 10 chains, the inference deteriorates and is outperformed by the LB2 and LB10 methods. This could be due to the parallel tempering scheme constraining the mismatch between the gradients in chains closer to the posterior, allowing for better estimates of the parameters.

7 Discussion

We have proposed a modification of a recently proposed gradient matching approach for systems biology (INF), and we have carried out a comparative evaluation of various state-of-the-art gradient matching methods. These methods are based on different statistical modelling and inference paradigms: non-parametric Bayesian statistics with Gaussian processes (INF, LB2, LB10), penalised regression splines (RAM), splines-based smooth functional tempering (C&S), and penalised likelihood based on reproducing kernel Hilbert spaces (GON). We have also compared the antagonistic paradigms of Bayesian inference (INF) versus parallel tempering (LB2, LB10) of slack parameters in the specific context of adaptive gradient matching. The GON method produces estimates that are close the true parameters in terms of absolute uncertainty. This however, was for the case with small observational noise (Gaussian iid noise sd $= 0.1$) and it would be interesting to see how the parameter estimation accuracy is affected by the increase of noise. The C&S method does well only in the one case, where the number of observations are very high (higher than what would be expected in these types of experiments) and the tuning parameters are finely adjusted (which in practise is very difficult and time-consuming). When the number of observations was reduced, all settings for this method deteriorated significantly. It is important also to note that the settings that we found to be optimal were slightly different than in the original paper, which highlights the sensitivity and unreliability in the splines based method. The INF method shows a reasonable performance in terms of consistently producing results close to the true parameters, across all the set-ups we have examined. However, this technique's decrease in uncertainty is at the expense of bias. The LB2 and LB10 methods show the best performance across the set-ups. The parameter

accuracy is unbiased across the different ODE models and the different settings of those models. The parallel tempering seems to be quite robust, performing similarly across the various set-ups. We have explored four different schedules for the parallel tempering scheme (as shown in table 3). Overall, the performance of parallel tempering has been found to be reasonably robust with respect to a variation of the schedule.

References

1. Calderhead, B., Girolami, M.A., Lawrence, N.D.: Accelerating Bayesian inference over non-linear differential equations with Gaussian processes. Neural Information Processing Systems (NIPS), 22 (2008)
2. Calderhead, B.: A study of Population MCMC for estimating Bayes Factors over nonlinear ODE models. University of Glasgow (2008)
3. Campbell, D., Steele, R.J.: Smooth functional tempering for nonlinear differential equation models. Stat. Comput. 22, 429–443 (2012)
4. Dondelinger, F., Filippone, M., Rogers, S., Husmeier, D.: ODE parameter inference using adaptive gradient matching with Gaussian processes. In: The 16th Internat. Conf. on Artificial Intelligence and Statistics (AISTATS). JMLR, vol. 31, pp. 216–228 (2013)
5. FitzHugh, R.: Impulses and physiological states in models of nerve membrane. Biophys. J. 1, 445–466 (1961)
6. Lawrence, N.D., Girolami, M., Rattray, M., Sanguinetti, G.: Learning and Inference in Computational Systems Biology. MIT Press, Cambridge (2010)
7. Lotka, A.: The growth of mixed populations: two species competing for a common food supply. Journal of the Washington Academy of Sciences 22, 461–469 (1932)
8. Nagumo, J.S., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating a nerve axon. Proc. Inst. Radio Eng. 50, 2061–2070 (1962)
9. Pokhilko, A., Fernandez, A.P., Edwards, K.D., Southern, M.M., Halliday, K.J., Millar, A.J.: The clock gene circuit in Arabidopsis includes a repressilator with additional feedback loops. Molecular Systems Biology 8, 574 (2012)
10. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press (2006)
11. González, J., Vujačić, I., Wit, E.: Inferring latent gene regulatory network kinetics. Statistical Applications in Genetics and Molecular Biology 12(1), 109–127 (2013)
12. Vyshemirsky, V., Girolami, M.A.: Bayesian ranking of biochemical system models. Bioinformatics 24(6), 833–839 (2008)
13. Ramsay, J.O., Hooker, G., Campbell, D., Cao, J.: Parameter estimation for differential equations: a generalized smoothing approach. J. R. Statist., 741–796 (2007)
14. Friel, N., Pettitt, A.N.: Marginal likelihood estimation via power posteriors. J. Royal Statist. Soc.: Series B (Statistical Methodology) 70, 589–607 (2008)
15. Liang, H., Wu, H.: Parameter Estimation for Differential Equation Models Using a Framework of Measurement Error in Regression Models. J. Am. Stat. Assoc., 1570–1583 (December 2008)
16. Solak, E., Murray-Smith, R., Leithead, W.E., Leith, D.J., Rasmussen, C.E.: Derivative observations in Gaussian Process Models of Dynamic Systems. Advances in Neural Information Processing Systems, 9–14 (2003)

17. Mohamed, L., Calderhead, B., Filippone, M., Christie, M., Girolami, M.: Population MCMC methods for history matching and uncertainty quantification. Comput Geosci., 423–436 (2012)
18. Calderhead, B., Girolami, M.: Estimating Bayes Factors via thermodynamic integration and population MCMC. Comp. Stat. & Data Analysis. 53, 4028–4045 (2009)
19. Murphy, K.P.: Machine Learning. A Probabilistic Perspective. MIT Press (2012)
20. Murray, I., Adams, R.: Slice sampling covariance hyperparameters of latent Gaussian models. Advances in Neural Information Processing Systems (NIPS) 23 (2010)