Fast Gaussian process based gradient matching for parameter identification in systems of nonlinear ODEs

Author 1
Institution 1

Author 2
Institution 2

Author 3
Institution 3

Abstract

Parameter identification and comparison of dynamical systems is a challenging task in many fields. Bayesian approaches based on Gaussian process regression over time-series data have been successfully applied to infer the parameters of a dynamical system without explicitly solving it. While the benefits in computational cost are well established, the theoretical foundation has been criticized in the past. We offer a novel interpretation which leads to a better understanding and improvements in state-of-the-art performance in terms of accuracy, robustness and a decrease in run time due to a more efficient setup for general nonlinear dynamical systems.

1 INTRODUCTION

The underlying mechanism of many processes in science and engineering can often be described by ordinary differential equations (ODE). While the form of dynamical systems, the ODEs, can often be derived using expert knowledge, the parameters are usually unknown, can not be directly measured and have to be estimated from empirical time series data. Since nonlinear ODEs typically do not have a closed form solution, standard methods for statistical inference require the computationally expensive numerical integration of the ODEs every time the parameters are changed (Calderhead et al., 2008).

To circumvent the high computational cost of numerical integration, gradient matching techniques have been proposed (e.g. Ramsay et al., 2007; Dondelinger et al., 2013; Niu et al., 2016; Gorbach et al., 2017a). Gradient matching is based on minimizing the difference between a model interpolating the dynamics of the state variables and the time derivatives provided by the ODEs. The first steps of this approach go back to spline-based methods, with an overview given by Ramsay et al. (2007). Calderhead et al. (2008) then proposed a fully probabilistic treatment by using Gaussian process models, increasing data efficiency and providing final uncertainty estimates. To match the gradients of the GP model to the gradients provided by the ODEs, Calderhead et al. (2008) introduce a product of experts heuristics (PoE). This heuristic has since become the state-of-the-art explanation, being reused, e.g., in Dondelinger et al. (2013) and Gorbach et al. (2017a). The advantages of the method have been well established, with applications in biomedical domains e.g., systems biology or neuroscience (Babtie et al., 2014; Macdonald and Husmeier, 2015). Given these applications, a solid understanding and theoretical framework of theses methods is of critical importance.

However, there are two important open issues. Regarding the theoretical framework, there has been some criticism regarding the use of the PoE in this context, e.g., by Wang and Barber (2014), who introduced a different modeling paradigm. Regarding empirical performance, Gorbach et al. (2017a) recently introduced a mean field variational scheme to decrease the run time. However, despite the expected run time savings, Gorbach et al. (2017a) also reported a significantly increased accuracy of the variational approach compared to the MCMC sampling based counterparts.

While the criticism of the product of experts approach (Wang and Barber, 2014) lead to the controversy of mechanistic modeling with Gaussian processes (Macdonald et al., 2015) where the theoretical inconsistencies of the modeling proposal of Wang and Barber (2014) have been outlined, a similar analysis investigating the theoretical foundation of the state of the art approach underlying all current modeling proposals (Calderhead et al., 2008; Dondelinger et al., 2013; Gorbach et al., 2017a) has been missing.

Preliminary work. Under review by AISTATS 2019. Do not distribute.

Our contributions. In this work, we

1. analyze the Product of Experts (PoE) heuristic, discovering and explaining theoretical inconsistencies of the state of the art approach,
2. provide a theoretical framework, replacing the criticized product of experts heuristic,
3. identify the cause of the performance gains of the variational approach of Gorbach et al. (2017a) over sampling-based methods,
4. combine these insights to create a novel algorithm improving on state-of-the-art performance for nonlinear systems in terms of accuracy and robustness, while providing a more computationally efficient sampling scheme reducing its run time by roughly 35%.[1]

2 PRELIMINARIES

2.1 Problem Formulation

In this work, we analyze an arbitrary system of parametric ordinary differential equations, whose state derivatives can be parametrized by a time independent parameter vector $\boldsymbol{\theta}$. In this setting, the true evolution of the dynamical system is given by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) \tag{1}$$

where $\mathbf{x}$ are the time dependent states of the system and $\mathbf{f}$ is an arbitrary, potentially nonlinear vector valued function.

While Equation (1) is meant to represent the dynamics of the system at all time points, $\mathbf{y}$, $\mathbf{x}$ and $\dot{\mathbf{x}}$ will be used throughout this paper to denote the vector containing the time evolution of one state or one state derivative at the observation times $\mathbf{t}$, i.e. $\mathbf{x} = [x_0(t_0), ..., x_0(t_N)]$ etc.

At N discrete time points $\mathbf{t}$, the state trajectories are observed with additive, i.i.d. Gaussian noise $\boldsymbol{\epsilon}(t_i) \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$, i.e.,

$$\mathbf{y}(t_i) = \mathbf{x}(t_i) + \boldsymbol{\epsilon}(t_i), \qquad i = 1 \dots \mathrm{N}, \tag{2}$$

or equivalently

$$p(\mathbf{y}|\mathbf{x}, \sigma) = \mathcal{N}(\mathbf{y}|\mathbf{x}, \sigma^2 \boldsymbol{I}). \tag{3}$$

Given these noisy observations $\mathbf{y}$ and the functional form of $\mathbf{f}$, the goal is to infer the true parameters $\boldsymbol{\theta}$.

For the sake of clarity, we present the theory for a one-dimensional system only and assume that all observations were created using one realization of the experiment. The extension to multidimensional systems (as done in the experiments, Section 5) or repeated experiments is straightforward but omitted to simplify the notation.

2.2 Modeling

The key idea of GP-based gradient matching is to build a GP regression model mapping the time points to the corresponding state values. For this, one needs to choose an appropriate kernel function $k_\phi(t_i, t_j)$, which is parametrized by the hyperparameters $\boldsymbol{\phi}$. Both, the choice of kernels as well as how to fit its hyperparameters is discussed in Section 7.3.

Once the kernel and its hyperparameters are fixed, the covariance matrix $\mathbf{C}_\phi$, whose elements are given by $\mathbf{C}_\phi(i,j) = k(t_i, t_j)$, can be constructed and used to define a standard zero mean Gaussian process prior on the states:

$$p(\mathbf{x}|\boldsymbol{\phi}) = \mathcal{N}(\mathbf{x}|\mathbf{0}, \mathbf{C}_\phi). \tag{4}$$

As differentiation is a linear operation, the derivative of a Gaussian process is again a Gaussian process. Using probabilistic calculus (see supplementary material, Section 7.1 for details), this fact leads to a distribution over the derivatives conditioned on the states at the observation points:

$$p(\dot{\mathbf{x}}|\mathbf{x}, \boldsymbol{\phi}) = \mathcal{N}(\dot{\mathbf{x}}|\mathbf{D}\mathbf{x}, \mathbf{A}). \tag{5}$$

Lastly, the information provided by the differential equation is used as well. For known states and parameters, one can calculate the derivatives using equation (1). A potential modeling mismatch between the output of the ODEs and the derivatives of the GP model is accounted for by introducing isotropic Gaussian noise with standard deviation γ, leading to the following Gaussian distribution over the derivatives:

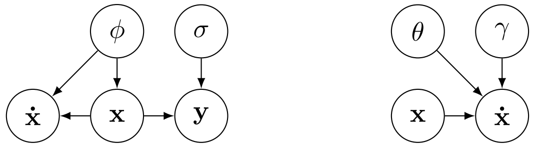
$$p(\dot{\mathbf{x}}|\mathbf{x}, \boldsymbol{\theta}, \gamma) = \mathcal{N}(\dot{\mathbf{x}}|f(\mathbf{x}, \boldsymbol{\theta}), \gamma^2 \boldsymbol{I}). \tag{6}$$

The modeling assumptions are summarized in the graphical models shown in Figure 1.

2.3 Inference

As stated in Section 2.1, the main goal of the inference process is to learn the parameters $\boldsymbol{\theta}$ using the noisy observations $\mathbf{y}$. Thus, it is necessary to connect the two graphical models shown in Figure 1. As shown in the previous section, $\mathbf{x}$ and the $\dot{\mathbf{x}}$ represent the same variables in both models. However, it is not straightforward to use this fact to combine the two. While

[1] Code publicly available at *Link removed to preserve author anonymity*



(a) Gaussian Process Model (b) ODE Model

Figure 1: Modeling assumptions of Gaussian Process based gradient matching.

it is intuitive to use the probability density over $\mathbf{x}$ of the Gaussian process model directly as the prior for $\mathbf{x}$ in the ODE response model, handling $\dot{\mathbf{x}}$ is more challenging. In both models, $\dot{\mathbf{x}}$ is a dependent variable. Thus, some heuristic is needed to combine the two conditional distributions $p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\phi})$ and $p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\theta},\gamma)$.

2.3.1 Product of experts heuristic

The main idea of the product of experts, originally introduced by Hinton (2002), is to infer the probability density of a variable by normalizing the product of multiple expert densities. Calderhead et al. (2008) use this to connect the two distributions over $\dot{\mathbf{x}}$, leading to

$$p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\phi},\boldsymbol{\theta},\gamma) \propto p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\phi})p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\theta},\gamma) \qquad (7)$$

The idea of this approach is that the resulting density only assigns high probability if both experts assign high probabilities. Hence, it considers only cases in which both experts agree. It is thus based on the intuition that the true $\boldsymbol{\theta}$ should correspond to a model that agrees both with the ODE model and the observed data. While this is intuitively well-motivated, we will show that the product of experts heuristic leads to theoretical difficulties and offer an alternative.

2.3.2 Markov Chain Monte Carlo based methods

Calderhead et al. (2008) combine the product of experts with equations (3), (4) and (5) and some suitable prior over $\boldsymbol{\theta}$ to obtain a joint distribution $p(\mathbf{x},\dot{\mathbf{x}},\boldsymbol{\theta},\boldsymbol{\phi},\sigma|\mathbf{y})$. After integrating out $\dot{\mathbf{x}}$, which can be done analytically since Gaussian processes are closed under linear operators (and using some proportionality arguments), a sampling scheme was derived that consists of two MCMC steps. First, the hyperparameters of the GP, $\boldsymbol{\phi}$ and σ, are sampled from the conditional distribution $p(\boldsymbol{\phi},\sigma|\mathbf{y})$. Then, a second MCMC scheme is deployed to infer the parameters of the ODE model, $\boldsymbol{\theta}$ and γ, by sampling from the conditional distribution $p(\boldsymbol{\theta},\gamma|\mathbf{x},\boldsymbol{\phi},\sigma)$. Dondelinger et al. (2013) then reformulated the approach by directly calculating the joint distribution

$$\begin{aligned} p(\mathbf{y},\mathbf{x},\boldsymbol{\theta},\boldsymbol{\phi},\gamma,\sigma) \propto\ & p(\boldsymbol{\theta}) \\ & \times \mathcal{N}(\mathbf{x}|\mathbf{0},\boldsymbol{C}_{\boldsymbol{\phi}}) \\ & \times \mathcal{N}(\mathbf{y}|\mathbf{x},\sigma^2\mathbf{I}) \\ & \times \mathcal{N}(\mathbf{f}(\mathbf{x},\boldsymbol{\theta})|\mathbf{D}\mathbf{x},\mathbf{A}+\gamma\mathbf{I}), \qquad (8) \end{aligned}$$

where the proportionality is meant to be taken w.r.t. the latent states $\mathbf{x}$ and the ODE parameters $\boldsymbol{\theta}$. Here $p(\boldsymbol{\theta})$ denotes some prior on the ODE parameters. This approach was named Adaptive Gradient Matching (AGM).

2.3.3 Variational inference

The main idea of Variational Gradient Matching (VGM), introduced by Gorbach et al. (2017a), is to substitute the MCMC inference scheme of AGM with a mean field variational inference approach, approximating the density in Equation (8) with a fully factorized Gaussian over the states $\mathbf{x}$ and the parameters $\boldsymbol{\theta}$. To obtain analytical solutions, the functional form of the ODEs is restricted to locally linear functions that could be written as

$$f(\mathbf{x},\boldsymbol{\theta}) = \sum_i \theta_i \prod_{j\in\mathcal{M}_i} x_j \qquad \text{where} \quad \mathcal{M}_i \subseteq \{1,...,K\}. \qquad (9)$$

As perhaps expected, VGM is magnitudes faster than the previous sampling approaches. However, despite being a variational approach, VGM was also able to provide significantly more accurate parameter estimates than both sampling-based approaches of Calderhead et al. (2008) and Dondelinger et al. (2013). In Section 4, we provide justification for these surprising performance differences.

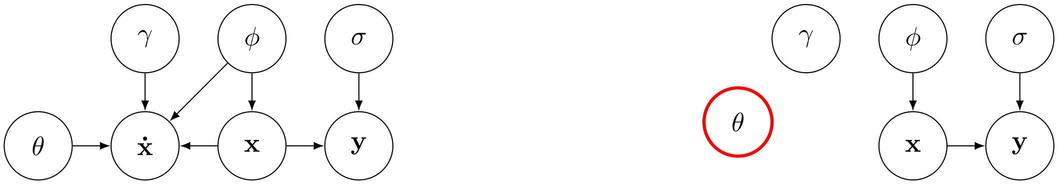
3 THEORY

3.1 Analysis Of The Product Of Experts Approach

As previously stated, Calderhead et al. (2008), Dondelinger et al. (2013) and Gorbach et al. (2017a) all use a product of experts to obtain $p(\dot{\mathbf{x}}|\mathbf{x},\boldsymbol{\phi},\boldsymbol{\theta},\gamma)$ as stated in Equation (7).

In this section, we will first provide an argument based on graphical models and then an argument based on the original mathematical derivation to illustrate challenges arising from this heuristic.

Figure 2 depicts what is happening if the product of experts approach is applied in the gradient matching framework. Figure 2a depicts the graphical model after the two models have been merged using the product



(a) After Product of Experts

(b) After Marginalizing $\dot{\mathbf{x}}$

Figure 2: The product of experts approach as a graphical model. After marginalization of $\dot{\mathbf{x}}$, the parameters $\boldsymbol{\theta}$ we would like to infer are independent of the observations $\mathbf{y}$.

of experts heuristic of Equation (7). Using the distribution over $\mathbf{x}$ of the Gaussian process model 1a as a prior for the $\mathbf{x}$ in the ODE response model 1b, effectively leads to merging the two nodes representing $\mathbf{x}$. Furthermore, the product of experts heuristic implies by its definition that after applying Equation (7), $\dot{\mathbf{x}}$ is only depending on $\mathbf{x}$, $\boldsymbol{\phi}$, $\boldsymbol{\theta}$ and γ.

In the graphical model in Figure 2a, the problem is already visible. The ultimate goal of merging the two graphical models is to create a probabilistic link between the observations $\mathbf{y}$ and the ODE parameters $\boldsymbol{\theta}$. However, the newly created connection between these two variables is given via $\dot{\mathbf{x}}$, which has no outgoing edges and of which no observations are available. Marginalizing out $\dot{\mathbf{x}}$ as proposed in the traditional approaches consequently leads to the graphical model in Figure 2b. As there is no directed path connecting other variables via $\dot{\mathbf{x}}$, all the different components are now independent. Consequently, the posterior over $\boldsymbol{\theta}$ is now given by the prior we put on $\boldsymbol{\theta}$ in the first place.

This problem can further be illustrated by the mathematical derivations in the original paper of Calderhead et al. (2008). After calculating the necessary normalization constants, the last equation in the third chapter is equivalent to stating

$$p(\boldsymbol{\theta}, \gamma | \mathbf{x}, \phi, \sigma) = \int p(\boldsymbol{\theta}) p(\gamma) p(\dot{\boldsymbol{x}} | \mathbf{x}, \boldsymbol{\theta}, \gamma, \phi, \sigma) d\dot{\mathbf{x}}. \quad (10)$$

It is clear that this equation should simplify to

$$p(\boldsymbol{\theta}, \gamma | \mathbf{x}, \phi, \sigma) = p(\boldsymbol{\theta}) p(\gamma). \quad (11)$$

Thus, one could argue that any links that are not present in the graphical model of Figure 2b but found by Calderhead et al. (2008) and reused in Dondelinger et al. (2013) and Gorbach et al. (2017a) were created by improper normalization of the density $p(\dot{\boldsymbol{x}} | \mathbf{x}, \boldsymbol{\theta}, \gamma, \phi, \sigma)$.

3.2 Adapting The Original Graphical Model

Despite these technical difficulties arising from the PoE heuristic, the approaches provide good empirical results and have been used in practice, e.g., by Babtie et al. (2014). In what follows, we derive an alternative model and mathematical justification for Equation (8) to provide a theoretical framework explaining the good empirical performance of Gaussian process-based gradient matching approaches, especially from Gorbach et al. (2017a), which uses only weak or nonexistent priors.

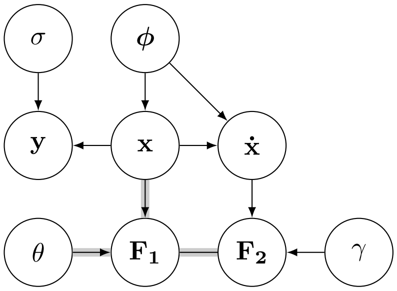


Figure 3: Alternative probabilistic model without PoE heuristic. Gray shaded connections are used to indicate a deterministic relationship.

The graphical model shown in Figure 3 offers an alternative approach to the product of experts heuristic. The top two layers are equivalent to a GP prior on the states, the induced GP on the derivatives and the observation model, as shown in Figure 1a.

The interesting part of the new graphical model is the bottom layer. Instead of adding a second graphical model like in Figure 1b to account for the ODE response, two additional random variables are introduced.

$\mathbf{F_1}$ is the deterministic output of the ODEs, assuming the values of $\mathbf{x}$ and $\boldsymbol{\theta}$ are given, i.e. $\mathbf{F_1} = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$. The deterministic nature of this equation is represented as a Dirac delta function:

$$p(\mathbf{F_1} | \mathbf{x}, \boldsymbol{\theta}) = \delta(\mathbf{F_1} - \mathbf{f}(\boldsymbol{\theta}, \mathbf{x})) \quad (12)$$

If the GP model were able to capture the true states and true derivatives perfectly, this new random variable should be equivalent to the derivatives of the GP, i.e., $\mathbf{F_1} = \dot{\mathbf{x}}$. However, to compensate for a potential model mismatch and slight errors of both GP states and GP derivatives, this condition is relaxed to

$$\mathbf{F_1} = \dot{\mathbf{x}} + \boldsymbol{\epsilon} =: \mathbf{F_2}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \gamma\mathbf{I}) \tag{13}$$

In the graphical model, this intuitive argument is encoded via the random variable $\mathbf{F_2}$. Given the $\dot{\mathbf{x}}$ provided by the GP model, Gaussian noise with standard deviation γ is added to create $\mathbf{F_2}$, whose probability density can thus be described as

$$p(\mathbf{F_2}|\dot{\mathbf{x}}, \gamma) = \mathcal{N}(\mathbf{F_2}|\dot{\mathbf{x}}, \gamma\mathbf{I}). \tag{14}$$

The equality constraint given by equation (13) is represented in the graphical model by the undirected edge between $\mathbf{F_1}$ and $\mathbf{F_2}$. When doing inference, this undirected edge is incorporated in the joint density via a Dirac delta function $\delta(\mathbf{F_2} - \mathbf{F_1})$. Thus, the joint density of the graphical model represented in Figure 3 can be written as

$$\begin{aligned} p(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{y}, &\mathbf{F_1}, \mathbf{F_2}, \boldsymbol{\theta}|\boldsymbol{\phi}, \sigma, \gamma) = p(\boldsymbol{\theta}) \\ &\times p(\mathbf{x}|\boldsymbol{\phi}) p(\dot{\mathbf{x}}|\mathbf{x}, \boldsymbol{\phi}) p(\mathbf{y}|\mathbf{x}, \sigma) \\ &\times p(\mathbf{F_1}|\boldsymbol{\theta}, \mathbf{x}) p(\mathbf{F_2}|\dot{\mathbf{x}}, \gamma\mathbf{I}) \delta(\mathbf{F_1} - \mathbf{F_2}). \end{aligned} \tag{15}$$

3.3 Inference In The New Model

Given all the definitions in the previous section, inference can now be directly performed without the need for additional heuristics. The result is a theoretically sound justification of the main result of Calderhead et al. (2008) and Dondelinger et al. (2013):

Theorem 1 *Given the modeling assumptions summarized in the graphical model in Figure 3,*

$$\begin{aligned} p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y}, \boldsymbol{\phi}, \gamma, \sigma) \propto\ & p(\boldsymbol{\theta}) \\ &\times \mathcal{N}(\mathbf{x}|\mathbf{0}, \boldsymbol{C_\phi}) \\ &\times \mathcal{N}(\mathbf{y}|\mathbf{x}, \sigma^2\mathbf{I}) \\ &\times \mathcal{N}(\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})|\mathbf{D}\mathbf{x}, \mathbf{A} + \gamma\mathbf{I}). \end{aligned} \tag{16}$$

The proof can be found in the supplementary material, section 7.2.

4 FAST GAUSSIAN PROCESS GRADIENT MATCHING

4.1 Hyperparameters

Using the theoretical framework of the previous section, it is clear that the performance of any algorithm based on equation 16 will heavily rely on the quality of the hyperparameters $\boldsymbol{\phi}$, as $\mathbf{C}_{\boldsymbol{\phi}}$, $\mathbf{D}$ and $\mathbf{A}$ are all depending on $\boldsymbol{\phi}$. In GP regression, it is common to fit the hyperparameters to the observations using a maximum likelihood scheme (see supplementary material, section (7.3)). However, neither VGM nor AGM actually do this.

In AGM, the hyperparameters are inferred concurrently to the states $\mathbf{x}$ and the ODE parameters $\boldsymbol{\theta}$ in one big MCMC scheme. Besides the need for clear hyperpriors on the hyperparameters and obvious drawbacks regarding running time, e.g. at each MCMC step $\mathbf{C}_{\boldsymbol{\theta}}$, $\mathbf{D}$ and $\mathbf{A}$ have to be recalculated and potentially inverted, this leads to a rough probability landscape requiring a complicated multi-chain setup (Dondelinger et al., 2013).

In VGM, this problem was sidestepped by treating the hyperparameters as tuning parameters that had to be set by hand. As these hyperparameters are not learned from data, it is obviously not a fair comparison. In our experiments, we thus used a maximum likelihood scheme to infer the hyperparameters before using the implementation of Gorbach et al. (2017a). To indicate this modification, the modified VGM algorithm was called MVGM.

4.2 FGPGM

However, MVGM is still outperforming AGM significantly as shown in the experiments in section 5. This suggests that the concurrent optimization of $\boldsymbol{\phi}$, $\mathbf{x}$ and $\boldsymbol{\theta}$ suggested by Dondelinger et al. (2013) is not helpful for the performance of the system. Based on this insight, we propose the sequential approach shown in Algorithm 1. In a first step, the Gaussian process model is fit to the standardized data by calculating the hyperparameters via equation (36). Then, the states $\mathbf{x}$ and ODE parameters $\boldsymbol{\theta}$ are inferred using a one chain MCMC scheme on the density given by equation (8).

5 EXPERIMENTS

For all experiments involving AGM, the R toolbox deGradInfer (Macdonald and Dondelinger, 2017) published alongside (Macdonald, 2017) was used. The toolbox was provided with code to run two experiments, namely Lotka Volterra and Protein Transduction. Both of these systems are used in this paper, as they are the two standard benchmark systems used in all previous publications. It should be noted however that the applicability of FGPGM is not restricted to these systems. Unlike AGM, FGPGM refrains from using hard to motivate hyperpriors and our publicly available implementation can easily be adapted to new settings.

For details about implementation and additional plots, refer to the supplementary material in section 7.6. It should be noted however that all algorithms were provided with one hyperparameter. While the toolbox of

Algorithm 1 FGPGM

```
Input: $\mathbf{y}, \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}), \gamma, N_{MCMC}, N_{burnin}, \mathbf{t}, \sigma_s, \sigma_p$
Step 1: Fit GP model to data
for all $k \in K$ do
    $\boldsymbol{\mu}_{y,k} \leftarrow \text{mean}(\mathbf{y}_k)$
    $\sigma_{y,k} \leftarrow \text{std}(\mathbf{y}_k)$
    $\tilde{\mathbf{y}} \leftarrow (\mathbf{y_k} - \boldsymbol{\mu}_k)/\sigma_{y,k}$
    Find $\boldsymbol{\phi}_k$ and $\sigma_k$ by maximizing
    $p(\tilde{\mathbf{y}}|\mathbf{t}, \boldsymbol{\phi}_k, \sigma_k)$ given by equation (36)
end for
Step 2: Infer $\mathbf{x}$ and $\boldsymbol{\theta}$ using MCMC
$\mathcal{S} \leftarrow \emptyset$
for $i = 1 \rightarrow N_{MCMC} + N_{burnin}$ do
    for each state and parameter do
        $\mathcal{T} \leftarrow \emptyset$
        Propose a new state or parameter value
        adding a zero mean Gaussian increment
        with standard deviation $\sigma_s$ or $\sigma_p$.
        Accept proposed value based on the
        density given by equation (8)
        Add current value to $\mathcal{T}$
    end for
    Add the mean of $\mathcal{T}$ to $\mathcal{S}$
end for
Discard the first $N_{burnin}$ samples of $\mathcal{S}$
Return the mean of $\mathcal{S}$
Return: $\mathbf{x}, \boldsymbol{\theta}$
```

AGM had to be provided with the true standard deviation of the observation noise, MVGM and AGM were provided with γ. The γ was determined by testing eight different values logarithmically spaced between 1 and 10^{-4} and comparing the results based on observation fit.

5.1 Lotka Volterra

The first system in question is the Lotka Volterra system originally proposed in Lotka (1978). It describes a two dimensional system whose dynamics are given by

$$\begin{aligned}\dot{x}_1(t) &= \quad \theta_1 x_1(t) - \theta_2 x_1(t) x_2(t)\\ \dot{x}_2(t) &= -\theta_3 x_2(t) + \theta_4 x_1(t) x_2(t)\end{aligned}$$

We reproduce the experimental setup of Gorbach et al. (2017a), i.e., the system was observed in the time interval $[0, 2]$ at 20 evenly spaced observation times. The true states were initialized with $[5, 3]$ and Gaussian noise with standard deviation 0.1 (low noise) and standard deviation 0.5 (high noise) was added to create the observations.

5.2 Protein Transduction

The second system to be analyzed is called Protein Transduction and was originally proposed in Vyshemirsky and Girolami (2008). It is known to be notoriously difficult to fit with unidentifiable parameters (Dondelinger et al., 2013). The dynamics are given by:

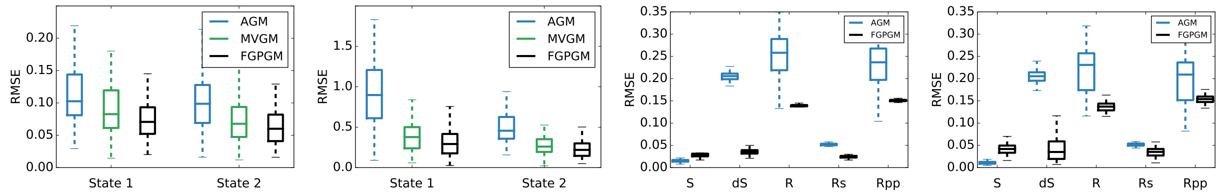
$$\begin{aligned}\dot{S} &= -\theta_1 S - \theta_2 SR + \theta_3 R_S\\ \dot{dS} &= \theta_1 S\\ \dot{R} &= -\theta_2 SR + \theta_3 R_S + \theta_5 \frac{R_{pp}}{\theta_6 + R_{pp}}\\ \dot{R_S} &= \theta_2 SR - \theta_3 R_S - \theta_4 R_S\\ \dot{R_{pp}} &= \theta_4 R_S - \theta_5 \frac{R_{pp}}{\theta_6 + R_{pp}} \qquad (17)\end{aligned}$$

It should be noted that these dynamics contain nonlinear terms violating the functional form assumption given in equation (9) of VGM inherited by MVGM. Nevertheless, both FGPGM and AGM can still be applied. For FGPGM, γ was set to 10^{-4}, while AGM was provided with the true observation noise standard deviations. The experimental setup of Dondelinger et al. (2013) was copied, i.e. the system was observed in the time interval $[0, 100]$ at the discrete observation times $\mathbf{t} = [0, 1, 2, 4, 5, 7, 10, 15, 20, 30, 40, 50, 60, 80, 100]$, the states were initialized with $\mathbf{x}(0) = [1, 0, 1, 0, 0]$ and the parameters were set to $\boldsymbol{\theta} = [0.07, 0.6, 0.05, 0.3, 0.017, 0.3]$. As in Dondelinger et al. (2013), Gaussian noise was added with standard deviation 0.001 (low noise) and 0.01 (high noise). As in the previous papers, a sigmoid kernel was used to deal with the logarithmically spaced observation times and the typically spiky form of the dynamics.

5.3 Evaluation

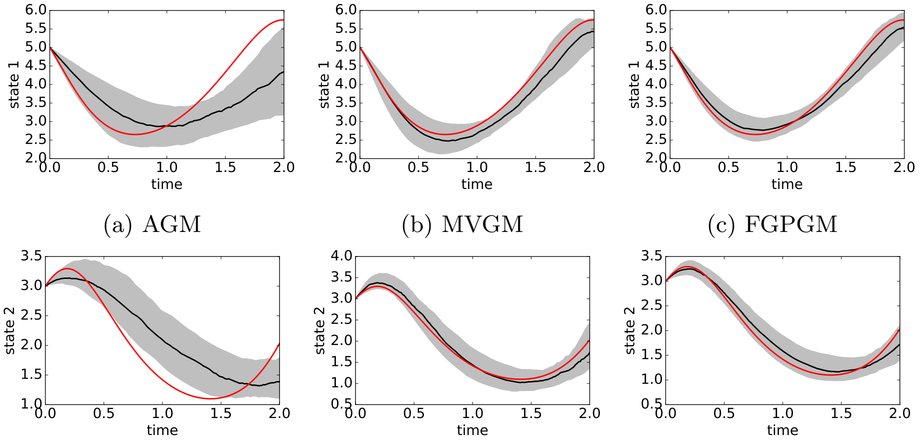
As the parameters of the Protein Transduction system are unidentifiable, comparing parameter values is not a good metric to rank approaches. Instead, after running the algorithms, we used a numerical integrator to obtain the trajectories corresponding to the dynamical system whose dynamics is given by the the inferred parameters. Then, the RMSE of these trajectories compared to the ground truth at the observation times were evaluated. For each experimental setting, this procedure was repeated for 100 different noise realizations. The results are shown in Figure 4. In Figure 5, we show median plots for the high noise setting of Lotka Volterra, while the running time between the state of the art AGM and FGPGM is shown in Figure 6.

While achieving run time savings of 35%, FGPGM



(a) LV low noise (b) LV high noise (c) PT low noise (d) PT high noise

Figure 4: RMSE of the trajectories obtained by numerical integration compared to the ground truth. Boxplot with median (line), 50% (box) and 75% (whisker) quantiles over 100 independent noise realizations. Lotka Volterra is shown on the left, Protein Transduction on the right.

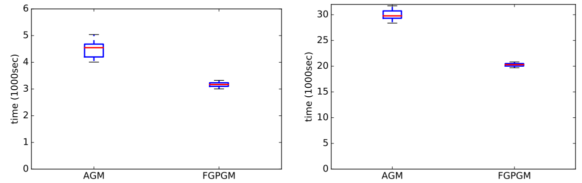


(a) AGM (b) MVGM (c) FGPGM

Figure 5: States after numerical integration of the inferred parameters in the high noise case of Lotka Volterra. Ground truth (red), median (black) and 75% quantiles (gray) over 100 independent noise realizations.

Table 1: Median reduction of state RMSE of FGPGM compared to AGM and VGM as baseline.

	AGM
LV low	35%
LV high	62%
PT low	50%
PT high	43%
	MVGM
LV low	13%
LV high	31%



(a) run time LV (b) run time PT

Figure 6: FGPGM shows a clearly reduced run time compared to AGM, saving roughly 35% of computation time.

shows an increased accuracy, reducing the state RMSE up to 62 % as shown in Table 1 and shows much lower variability across noise realizations. The effect is especially striking for the nonlinear system Protein Transduction, where the variance of the states R and R_{pp} has been reduced by at least one order of magnitude, while reducing the bias by more than 30%, see Figures 4c and 4d.

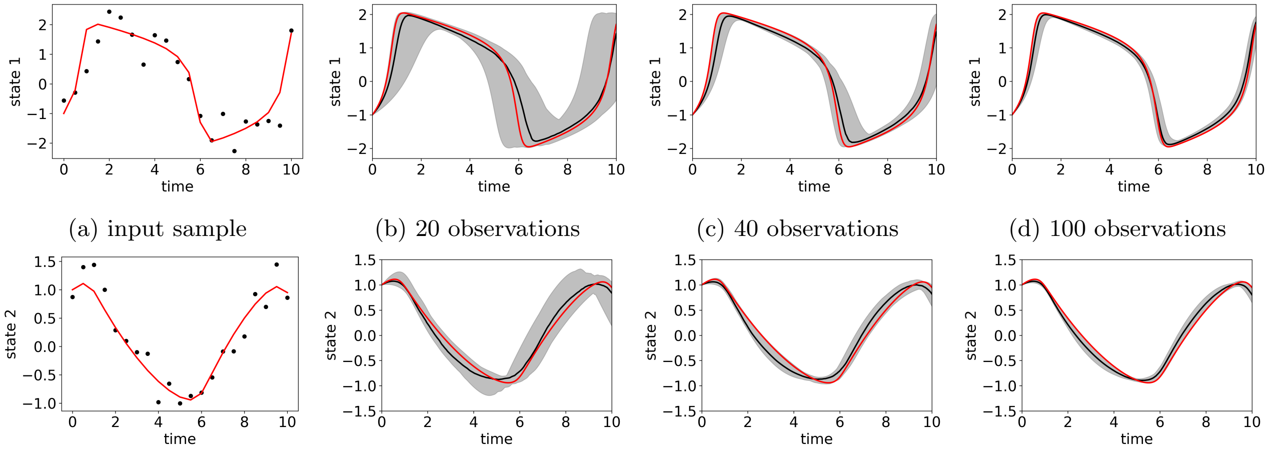
5.4 Spiky Dynamics

As can be seen in Figure 5, all GP based gradient matching algorithms converge to parameter settings where the trajectories are smoother than the ground truth. While learning the hyperparameters in a preprocessing step clearly reduces this effect for FGPGM and MVGM, there is still some bias. If only few observations are available, the GP prior on the states tends to smooth out part of the system dynamics. This "smoothing bias" is then passed on to the ODE parameters in the gradient matching scheme.

To investigate the practical importance of this effect, we evaluate FGPGM on a third system proposed by FitzHugh (1961) and Nagumo et al. (1962) for modeling giant squid neurons. Abbreviating the name of its inventors, we will refer to it as the FHN system. Its dynamics are given by the two ODEs

$$\dot{V} = \theta_1(V - \frac{V^3}{3} + R) \qquad (18)$$

$$\dot{R} = \frac{1}{\theta_1}(V - \theta_2 + \theta_3 R) \qquad (19)$$



(a) input sample

(b) 20 observations

(c) 40 observations

(d) 100 observations

Figure 7: One input sample and median plots of the numerically integrated states after parameter inference for the FHN system with SNR 10. Ground truth (red), median (black) and 75% quantiles (gray) over 100 independent noise realizations. In the most left plots, the black dots represent the noisy observations.

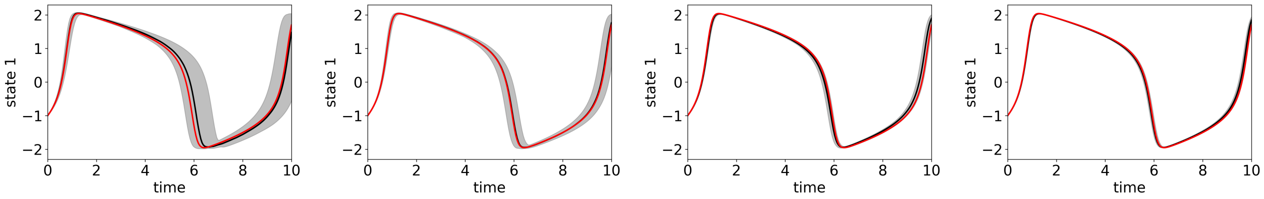


Figure 8: Median plots of the first state of FHN with SNR 100 in case of 10, 25, 50 and 100 observations. Same colors as in Figure 7. Plots of state 2 can be found in the supplementary material.

Due to its highly nonlinear terms, the FHN system has notoriously fast changing dynamics. One example realization including noisy observations is shown in Figure 7. To account for the spikier behavior of the system, we used a Matérn52 kernel. γ was set to $3 * 10^{-4}$.

While the bias towards smoother trajectories is clearly visible, FGPGM finds parameters that tightly hug the ground truth. As to be expected, the smoothing bias gets smaller if we add more observations. Furthermore, increasing the SNR to 100 as shown in Figure 8 leads to an impressive accuracy, even if we reduce the amount of observations to just 10, especially as FGPGM is a hyper-prior free, statistical method

6 DISCUSSION

Gradient matching is a successful tool to circumvent the computational cost of numerical integration for Bayesian parameter identification in dynamical systems, especially if the dynamics, like in most real world systems, are reasonably smooth. Previous Gaussian process-based approaches used a criticized product of experts heuristics, which leads to technical difficulties. We illustrated these theoretical problems and provided a novel, sound formulation that does not rely on a PoE. We furthermore explained the surprising performance gains of variational over sampling-based approaches and then combined these insights to propose a new algorithm called FGPGM, which jointly learns states and parameters with improved state-of-the-art performance in terms of accuracy, run time, robustness and reduction of ”smoothing bias” for general nonlinear dynamical systems.

Unlike the previous MCMC approaches, FGPGM uses a one-chain Metropolis Hastings scheme, which is much easier to tune than the previously used, complicated multi chain setups. Due to the sequential fitting of the hyperparameters, the wild behavior of the probability density motivating the setup in Dondelinger et al. (2013, section 3) and the need for hard to motivate hyperpriors is successfully avoided. Consequently, the inference is significantly more accurate, faster, more robust and significantly decreases the smoothing bias of GP based gradient matching schemes.

References

Ann C Babtie, Paul Kirk, and Michael PH Stumpf. Topological sensitivity analysis for systems biology. *Proceedings of the National Academy of Sciences*, 111(52):18507–18512, 2014.

Stefan Bauer, Nico S Gorbach, Djordje Miladinovic, and Joachim M. Buhmann. Efficient and flexible inference for stochastic systems. *Neural Information Processing Systems (NIPS)*, 2017.

Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.

Ben Calderhead, Mark Girolami and Neil D. Lawrence. Accelerating bayesian inference over nonliner differential equations with gaussian processes. *Neural Information Processing Systems (NIPS)*, 2008.

Taeryon Choi and Mark J Schervish. On posterior consistency in nonparametric regression problems. *Journal of Multivariate Analysis*, 98(10):1969–1987, 2007.

Frank Dondelinger, Maurizio Filippone, Simon Rogers and Dirk Husmeier. Ode parameter inference using adaptive gradient matching with gaussian processes. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2013.

David Duvenaud. *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge, 2014.

David Duvenaud, James Robert Lloyd, Roger Grosse, Joshua B Tenenbaum, and Zoubin Ghahramani. Structure discovery in nonparametric regression through compositional kernel search. *arXiv preprint arXiv:1302.4922*, 2013.

Richard FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal*, 1(6):445–466, 1961.

Javier González, Ivan Vujačić, and Ernst Wit. Reproducing kernel hilbert space based estimation of systems of ordinary differential equations. *Pattern Recognition Letters*, 45:26–32, 2014.

Nico S Gorbach, Stefan Bauer, and Joachim M. Buhmann. Scalable variational inference for dynamical systems. *Neural Information Processing Systems (NIPS)*, 2017a.

Nico S Gorbach, Andrew An Bian, Benjamin Fischer, Stefan Bauer, and Joachim M Buhmann. Model selection for gaussian process regression. In *German Conference on Pattern Recognition*, pages 306–318. Springer, 2017b.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Alfred J Lotka. The growth of mixed populations: two species competing for a common food supply. In *The Golden Age of Theoretical Ecology: 1923–1940*, pages 274–286. Springer, 1978.

Benn Macdonald. *Statistical inference for ordinary differential equations using gradient matching*. PhD thesis, University of Glasgow, 2017.

Benn Macdonald and Frank Dondelinger. deGradInfer: Parameter inference for systems of differential equation. `https://CRAN.R-project.org/package=deGradInfer`, 2017.

Benn Macdonald and Dirk Husmeier. Gradient matching methods for computational inference in mechanistic models for systems biology: a review and comparative analysis. *Frontiers in bioengineering and biotechnology*, 3, 2015.

Benn Macdonald, Catherine F. Higham and Dirk Husmeier. Controversy in mechanistic modemodel with gaussian processes. *International Conference on Machine Learning (ICML)*, 2015.

Jinichi Nagumo, Suguru Arimoto, and Shuji Yoshizawa. An active pulse transmission line simulating nerve axon. *Proceedings of the IRE*, 50 (10):2061–2070, 1962.

Mu Niu, Simon Rogers, Maurizio Filippone, and Dirk Husmeier. Fast inference in nonlinear dynamical systems using gradient matching. *International Conference on Machine Learning (ICML)*, 2016.

Athanasios Papoulis and S Unnikrishna Pillai. *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

Kaare Brandt Petersen, Michael Syskind Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7(15):510, 2008.

Jim O Ramsay, Giles Hooker, David Campbell, and Jiguo Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

Vladislav Vyshemirsky and Mark A Girolami. Bayesian ranking of biochemical system models. *Bioinformatics*, 24(6):833–839, 2008.

Yali Wang and David Barber. Gaussian processes for bayesian estimation in ordinary differential equations. *International Conference on Machine Learning (ICML)*, 2014.

Xinxin Zhang, Ruirui Ji, and Yingmin Yi. Inference of transcriptional regulation using multi-objective optimization with gaussian process. In *Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), International Congress on*, pages 1820–1825. IEEE, 2016.

7 Supplementary Material

7.1 Derivatives of a Gaussian process

7.1.1 Definitions

Following Papoulis and Pillai (2002), we define stochastic convergence and stochastic differentiability.

Definition 2 *The RV x_n* ***converges*** *to x in the MS sense (limit in mean) if for some x*

$$\lim_{n \to \infty} \mathbb{E}(|x_n - x|) = 0 \tag{20}$$

Definition 3 *The stochastic process x(t) is MS* ***differentiable*** *if for some $x'(t)$*

$$\lim_{\delta t \to 0} \mathbb{E} \left| \frac{x(t + \delta t) - x(t)}{\delta t} - x'(t) \right| = 0 \tag{21}$$

Definition 4 *A stochastic process x(t) is called a* ***Gaussian Process****, if any finite number of samples of its trajectory are jointly Gaussian distributed according to a previously defined mean function $\mu(t)$ and a covariance matrix, that can be constructed using a predefined kernel function $k_\phi(t_i, t_j)$*

7.1.2 GP and its derivative are jointly Gaussian

Let $t_0, \delta t \in \mathbb{R}$.
Let $x(t)$ be a Gaussian Process with constant mean μ and kernel function $k_\phi(t_1, t_2)$, assumed to be MS differentiable.

From the definition of GP, we know that $x(t_0 + \delta t)$ and $x(t_0)$ are jointly Gaussian distributed.

$$\begin{bmatrix} x(t_0) \\ x(t_0 + \delta t) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu \\ \mu \end{bmatrix}, \mathbf{\Sigma} \right) \tag{22}$$

where $\mathbf{\Sigma}_{i,j} = k_\phi(\mathbf{t}_i, \mathbf{t}_j)$ using $\mathbf{t} = [t_0, t_0 + \delta t]$.

Using the linear transformation

$$\mathbf{T} = \frac{1}{\delta t} \begin{bmatrix} \delta t & 0 \\ -1 & 1 \end{bmatrix} \tag{23}$$

one can show that

$$\begin{bmatrix} x(t_0) \\ \frac{x(t_0 + \delta t) - x(t_0)}{\delta t} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu \\ 0 \end{bmatrix}, \mathbf{T \Sigma T}^T \right) \tag{24}$$

So it is clear that for all δt, $x(t_0)$ and $\frac{x(t_0+\delta t) - x(t_0)}{\delta t}$ are jointly Gaussian distributed. Using the assumption that x is differentiable according to the definition in eq. 21 and the fact that convergence in expectation implies convergence in distribution, it is clear that $x(t_0)$ and $\dot{x}(t_0)$ are jointly Gaussian.

This fact can be easily extended to any finite set of sample times $\mathbf{t} = [t_0, t_1, ..., t_N]$. One can use the exact same procedure to show that the resulting vectors $\mathbf{x}(\mathbf{t})$ and $\dot{\mathbf{x}}(\mathbf{t})$ are jointly Gaussian as well.

7.1.3 Moments of the joint distribution

As shown in the previous section, any finite set of samples $\mathbf{x}$ is jointly Gaussian together with its derivatives $\dot{\mathbf{x}}$. To calculate the full distribution, it thus suffices to calculate the mean and the covariance between the elements of the full vector

$$\begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{C}_\phi & \mathbf{C}'_\phi \\ {}'\mathbf{C}_\phi & \mathbf{C}''_\phi \end{bmatrix} \right) \tag{25}$$

$\mathbf{C}_\phi$ is the predefined kernel matrix of the Gaussian Process.

${}'\mathbf{C}_\phi$ can be calculated by directly using the linearity of the covariance operator.

$$\begin{aligned} {}'\mathbf{C}_{\phi_{i,j}} &= \mathrm{cov}(\dot{x}(t_i), x(t_j)) \\ &= \mathrm{cov}\left(\frac{d}{da} x(a) \Big|_{a=t_i}, x(t_j) \right) \\ &= \frac{d}{da} \mathrm{cov}\left(x(a), x(t_j) \right)_{a=t_i} \\ &= \frac{d}{da} k_\phi(a, t_j)_{a=t_i} \end{aligned} \tag{26}$$

Obviously, $\mathbf{C}'_\phi$ is just the transposed of ${}'\mathbf{C}_\phi$, while $\mathbf{C}''_\phi$ can be calculated in exactly the same manner to obtain

$$\mathbf{C}''_{\phi_{i,j}} = \frac{d}{da} \frac{d}{db} k_\phi(a, b)_{a=t_i, b=t_j} \tag{27}$$

7.1.4 Conditional GP for derivatives

To obtain the GP over the derivatives given the states, the joint distribution

$$\begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{C}_\phi & \mathbf{C}'_\phi \\ {}'\mathbf{C}_\phi & \mathbf{C}''_\phi \end{bmatrix} \right) \tag{28}$$

has to be transformed. This can be done using standard techniques as described e.g. in section 8.1.3 of Petersen et al. (2008). There, it is written:

Define

$$\mathbf{x} = \begin{bmatrix} \mathbf{x_a} \\ \mathbf{x_b} \end{bmatrix}, \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu_a} \\ \boldsymbol{\mu_b} \end{bmatrix}, \mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma_a} & \mathbf{\Sigma_c} \\ \mathbf{\Sigma_c}^T & \mathbf{\Sigma_b} \end{bmatrix} \tag{29}$$

Then

$$p(\mathbf{x_b} | \mathbf{x_a}) \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_b, \hat{\mathbf{\Sigma}}_b) \tag{30}$$

where

$$\hat{\boldsymbol{\mu}}_b = \boldsymbol{\mu}_{\boldsymbol{b}} + {\boldsymbol{\Sigma}_{\boldsymbol{c}}}^T {\boldsymbol{\Sigma}_{\boldsymbol{a}}}^{-1}(\mathbf{x}_{\mathbf{a}} - \boldsymbol{\mu}_{\boldsymbol{a}}) \tag{31}$$

$$\hat{\boldsymbol{\Sigma}}_b = \boldsymbol{\Sigma}_{\boldsymbol{b}} - {\boldsymbol{\Sigma}_{\boldsymbol{c}}}^T {\boldsymbol{\Sigma}_{\boldsymbol{a}}}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{c}} \tag{32}$$

Applied to the above probability distribution, this leads to

$$p(\dot{\mathbf{x}}|\mathbf{x}) \sim \mathcal{N}(\mathbf{D}\mathbf{x}, \mathbf{A}) \tag{33}$$

using

$$\mathbf{D} = {'\mathbf{C}_\phi} {\mathbf{C}_\phi}^{-1} \tag{34}$$

$$\mathbf{A} = \mathbf{C}''_\phi - {'\mathbf{C}_\phi} {\mathbf{C}_\phi}^{-1} \mathbf{C}'_\phi \tag{35}$$

7.2 Proof of theorem 1

Proof The proof of this statement follows directly by combining all the previous definitions and marginalizing out all the random variables that are not part of the end result.

First, one starts with the joint density over all variables as stated in equation (15)

$$p(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{y}, \mathbf{F_1}, \mathbf{F_2}, \boldsymbol{\theta}|\boldsymbol{\phi}, \sigma, \gamma) = \\ p_{\text{GP}}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{y}|\boldsymbol{\phi}, \sigma) p_{\text{ODE}}(\mathbf{F_1}, \mathbf{F_2}, \boldsymbol{\theta}|\mathbf{x}, \dot{\mathbf{x}}, \gamma),$$

where

$$p_{\text{GP}}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{y}|\boldsymbol{\phi}, \sigma) = p(\mathbf{x}|\boldsymbol{\phi}) p(\dot{\mathbf{x}}|\mathbf{x}, \boldsymbol{\phi}) p(\mathbf{y}|\mathbf{x}, \sigma)$$

and

$$p_{\text{ODE}}(\mathbf{F_1}, \mathbf{F_2}, \boldsymbol{\theta}|\mathbf{x}, \dot{\mathbf{x}}, \gamma) = \\ p(\boldsymbol{\theta}) p(\mathbf{F_1}|\boldsymbol{\theta}, \mathbf{x}) p(\mathbf{F_2}|\dot{\mathbf{x}}, \gamma\mathbf{I}) \delta(\mathbf{F_1} - \mathbf{F_2}).$$

To simplify this formula, p_{ODE} can be reduced by marginalizing out $\mathbf{F_2}$ using the properties of the Dirac delta function and the probability density defined in equation (14). The new p_{ODE} is then independent of $\mathbf{F_2}$.

$$p_{\text{ODE}}(\mathbf{F_1}, \boldsymbol{\theta}|\mathbf{x}, \dot{\mathbf{x}}, \gamma) = p(\boldsymbol{\theta}) p(\mathbf{F_1}|\boldsymbol{\theta}, \mathbf{x}) \mathcal{N}(\mathbf{F_1}|\dot{\mathbf{x}}, \gamma\mathbf{I}).$$

Inserting equation (13) yields

$$p_{\text{ODE}}(\mathbf{F_1}, \boldsymbol{\theta}|\mathbf{x}, \dot{\mathbf{x}}, \gamma) = p(\boldsymbol{\theta}) \delta(\mathbf{F_1} - \mathbf{f}(\mathbf{x}, \boldsymbol{\theta})) \mathcal{N}(\mathbf{F_1}|\dot{\mathbf{x}}, \gamma\mathbf{I}).$$

Again, the properties of the Dirac delta function are used to marginalize out $\mathbf{F_1}$. The new p_{ODE} is now independent of $\mathbf{F_1}$,

$$p_{\text{ODE}}(\boldsymbol{\theta}|\mathbf{x}, \dot{\mathbf{x}}, \gamma) = p(\boldsymbol{\theta}) \mathcal{N}(\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})|\dot{\mathbf{x}}, \gamma\mathbf{I}).$$

This reduced p_{ODE} is now combined with p_{GP}. Observing that the mean and the argument of a normal density are interchangeable and inserting the definition of the GP prior on the derivatives given by equation (5) leads to

$$p(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{y}, \boldsymbol{\theta}|\boldsymbol{\phi}, \sigma, \gamma) = \\ p(\boldsymbol{\theta}) p(\mathbf{x}|\boldsymbol{\phi}) \mathcal{N}(\dot{\mathbf{x}}|\mathbf{D}\mathbf{x}, \mathbf{A}) p(\mathbf{y}|\mathbf{x}, \sigma) \mathcal{N}(\dot{\mathbf{x}}|\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}), \gamma\mathbf{I}).$$

$\dot{\mathbf{x}}$ can now be marginalized by observing that the product of two normal densities of the same variable is again a normal density. The formula can be found, e.g., in Petersen et al. (2008). As a result, one obtains

$$p(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta}|\boldsymbol{\phi}, \sigma, \gamma) = \\ p(\boldsymbol{\theta}) p(\mathbf{x}|\boldsymbol{\phi}) p(\mathbf{y}|\mathbf{x}, \sigma) \mathcal{N}(\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})|\mathbf{D}\mathbf{x}, \mathbf{A} + \gamma\mathbf{I}).$$

It should now be clear that after inserting equations (3) and (4) and renormalizing, we get the final result

$$p(\mathbf{x}, \boldsymbol{\theta}|\mathbf{y}, \boldsymbol{\phi}, \gamma, \sigma) \propto \\ p(\boldsymbol{\theta}) \mathcal{N}(\mathbf{x}|\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\phi}}) \mathcal{N}(\mathbf{y}|\mathbf{x}, \sigma^2\mathbf{I}) \mathcal{N}(\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})|\mathbf{D}\mathbf{x}, \mathbf{A} + \gamma\mathbf{I}),$$

concluding the proof of this theorem. □

7.3 Hyperparameter and kernel selection

As discussed before, the Gaussian process model is defined by a kernel function $k_\phi(t_i, t_j)$. For both the hyperparameters $\boldsymbol{\phi}$ and the functional form of k there exist many possible choices. Even though the exact choice might not be too important for consistency guarantees in GP regression (Choi and Schervish, 2007), this choice directly influences the amount of observations that are needed for reasonable performance. While there exist some interesting approaches to learn the kernel directly from the data, e.g., Duvenaud et al. (2013) and Gorbach et al. (2017b), these methods can not be applied due to the very low amount of observations of the systems considered in this paper. As in previous approaches, the kernel functional form is thus restricted to simple kernels with few hyperparameters, whose behaviors have already been investigated by the community, e.g., in the kernel cookbook by Duvenaud (2014). Once a reasonable kernel is chosen, it is necessary to fit the hyperparameter and depending on the amount of expert knowledge available, there are different methodologies.

7.3.1 Maximizing the data likelihood

As mentioned e.g. in Rasmussen (2004), it is possible for a Gaussian process model to analytically calculate the marginal likelihood of the observations $\mathbf{y}$ given the evaluation times $\mathbf{t}$ and hyperparameters $\boldsymbol{\phi}$ and σ.

$$\begin{aligned}\log(p(\mathbf{y}|\mathbf{t}, \boldsymbol{\phi}, \sigma)) = \\ &- \frac{1}{2}\mathbf{y}^T(\mathbf{C}_\phi + \sigma\mathbf{I})^{-1} \\ &- \frac{1}{2}\log|\mathbf{C}_\phi + \sigma\mathbf{I}| \\ &- \frac{n}{2}\log 2\pi \end{aligned} \tag{36}$$

where σ is the GPs estimate for the standard deviation of the observation noise and n is the amount of observations.

This equation is completely independent of the ODE system one would like to analyze and depends only on the observations of the states. To fit the GP model to the data, equation (36) can be maximized w.r.t. $\boldsymbol{\phi}$ and σ, without incorporating any prior knowledge.

7.3.2 Concurrent optimization

In AGM of Dondelinger et al. (2013), the hyperparameters are not calculated independent of the ODEs. Instead, a prior is defined and their posterior distribution is determined simultaneously with the posterior distribution over states and parameters by sampling from equation (8).

This approach has several drawbacks. As we shall see in section 5, its empirical performance is significantly depending on the hyperpriors. Furthermore, optimizing the joint distribution given equation (8) requires calculating the inverse of the covariance matrices $\mathbf{C}_\phi$ and $\mathbf{A}$, which has to be done again and again for each new set of hyperparameters. Due to the computational complexity of matrix inversion, this is significantly slowing down the optimization.

For these reasons, if strong prior knowledge about the hyperparameters is available, it might be better to incorporate it into the likelihood optimization presented in the previous section. There, it could be used as a hyperprior to regularize the likelihood.

7.3.3 Manual tuning

In the variational inference approach Gorbach et al. (2017a), the hyperparameters were assumed to be provided by an expert. If such expert knowledge is available, it should definitely be used since it can improve the accuracy drastically.

7.4 Adjusting the GP model

To make VGM more comparable to AGM, the hyperparameters of the kernel must be learned from the data. However, maximizing the data likelihood described in equation (36) directly using the prior defined in equation (4) will lead to very bad results.

7.4.1 Zero mean observations

The main reason for the poor performance without any pretreatment of $\mathbf{y}$ is the fact that the zero mean assumption in equation (36) is a very strong regularization for the amount of data available. As its effect directly depends on the distance of the true values to zero, it will be different for different states in multidimensional systems, further complicating the problem. Thus, it is common in GP regression to manipulate the observations such that they have zero mean.

This procedure can be directly incorporated into the joint density given by equation (8). It should be noted that for multidimensional systems this joint density will factorize over each state k, whose contribution will be given by

$$\begin{aligned} p(\mathbf{x_k}|\mathbf{y_k}, \boldsymbol{\theta}, \boldsymbol{\phi}, \gamma, \sigma) \propto &\mathcal{N}(\tilde{\mathbf{x}}_\mathbf{k}|\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\phi}_\mathbf{k}}) \\ &\times \mathcal{N}(\mathbf{y_k}|\mathbf{x_k}, \sigma^2\mathbf{I}) \\ &\times \mathcal{N}(\mathbf{f}_k(\mathbf{x_k}, \boldsymbol{\theta})|\mathbf{D_k}\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{A_k} + \gamma\mathbf{I}) \end{aligned} \tag{37}$$

where

$$\tilde{\mathbf{x}}_\mathbf{k} = \mathbf{x_k} - \mu_{y,k}\mathbb{1}$$

using $\mu_{y,k}$ to denote the mean of the observations of the k-th state and $\mathbb{1}$ to denote a vector of ones with appropriate length.

It is important to note that this transformation is not just equivalent to exchanging $\mathbf{x_k}$ and $\tilde{\mathbf{x}}_\mathbf{k}$. While the transformation is not necessary for the observation term, as $\mathbf{x_k}$ and $\mathbf{y_k}$ would be shifted equally, the original $\mathbf{x_k}$ is needed as input to the ODEs. This allows for this transformation without the need to manually account for this in the structural form of the differential equations.

This trick will get rid of some of the bias introduced by the GP prior. In the simulations, this made a difference for all systems, including the most simple one presented in section 5.1.

7.4.2 Standardized states

If the systems get more complex, the states might be magnitudes apart from each other. If one were to use the same hyperparameters $\boldsymbol{\phi}$ for all states, then a deviation $(\mathbf{F_k} - \mathbf{D_k}\tilde{\mathbf{x}}_\mathbf{k}) = 10^{-4}$ would contribute equally to the change in probability, independent of whether the states $\tilde{\mathbf{x}}_\mathbf{k}$ are of magnitude 10^{-8} or 10^3. Thus, small relative deviations from the mean of states with large values will lead to stronger changes in the joint probability than large relative deviations of states with small values. This is not a desirable property, which can be partially alleviated by calculating a new set of hyperparameters for each state. However, this problem can be completely nullified by standardizing the data $\mathbf{y}$. For the k-th state, this would change its contribution to the joint density to

$$p(\mathbf{x_k}|\mathbf{y_k}, \boldsymbol{\theta}, \boldsymbol{\phi}, \gamma, \sigma) \propto \mathcal{N}\left(\frac{1}{\sigma_{y,k}}\tilde{\mathbf{x}}_\mathbf{k}\middle|\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\phi}_\mathbf{k}}\right) \tag{38}$$
$$\times \mathcal{N}\left(\frac{1}{\sigma_{y,k}}\mathbf{y_k}\middle|\frac{1}{\sigma_{y,k}}\mathbf{x_k}, \sigma^2\mathbf{I}\right)$$
$$\times \mathcal{N}\left(\frac{1}{\sigma_{y,k}}\mathbf{f}_k(\mathbf{x_k}, \boldsymbol{\theta})\middle|\frac{1}{\sigma_{y,k}}\mathbf{D_k}\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{A_k} + \gamma\mathbf{I}\right)$$

where

$$\tilde{\mathbf{x}}_\mathbf{k} = \mathbf{x_k} - \mu_{y,k}\mathbb{1}$$

and $\sigma_{y,k}$ is the standard deviation of the observations of the k-th state.

For complex systems with states on different orders of magnitude, standardization is a must to obtain reasonable performance. Even for the system presented in section 5.2, standardization has a significantly beneficial effect, although the states do not differ greatly in magnitude.

7.5 Algorithmic details

For all experiments involving AGM, the R toolbox deGradInfer (Macdonald and Dondelinger, 2017) published alongside Macdonald (2017) was used. For comparability, no priors were used, but the toolbox needed to be supplied with a value for the standard deviation of the observational noise and the true standard deviation of the noise was used. For all other parameters, e.g., amount of chains or samples, the values reported in Dondelinger et al. (2013) were used.

For MVGM and FGPGM, the hyperparameters of the GP model were determined in a preprocessing step identical for both algorithms. After calculating the hyperparameters, the MVGM parameters were inferred using the implementation used by Gorbach et al. (2017a).

Both MVGM and FGPGM need to be supplied with γ, which was treated as a tuning parameter. In principle, this parameter could be found by evaluating multiple candidates in parallel and choosing based on data fit.

For both experiments, the amount of iterations recommended by the AGM toolbox (Macdonald and Dondelinger, 2017) have been used, namely 100'000 iterations for Lotka Volterra and 300'000 iterations for Protein Transduction. This is the same setup that has been used to obtain the parameter estimates shown throughout the paper. The simpler sampling setup of FGPGM clearly leads to running time savings of about a third compared to AGM. Thus, FGPGM is not only significantly more accurate, but also significantly faster than AGM. Dondelinger et al. (2013) have shown that this also implies order of magnitude improvements if compared to the running time of approaches based on numerical integration.

All experiments were performed using the *CLUSTER NAME REMOVED FOR BLIND SUBMISSION* cluster. It should be noted that the algorithms were implemented in different programming languages and the cluster consists of cores with varying computation power, adding some variability to the running time estimates.

7.6 Lotka Volterra

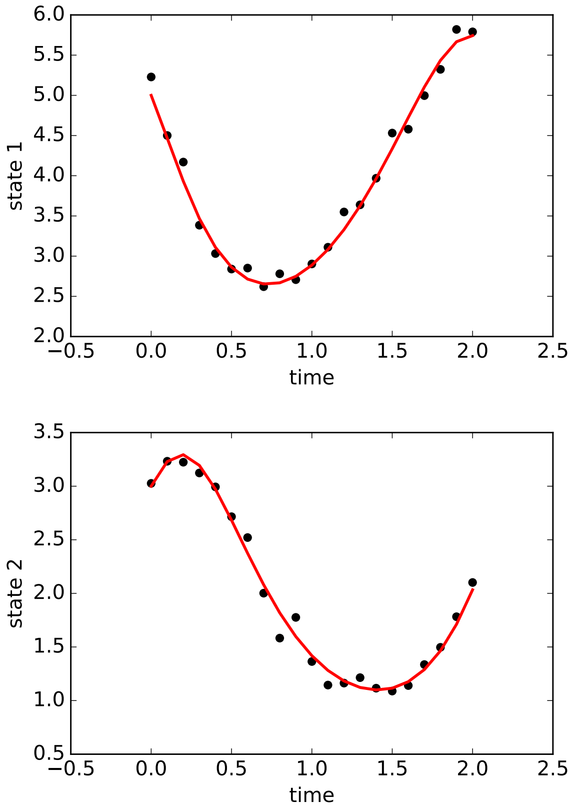


Figure 9: Example rollout of the Lotka Volterra system showing the state evolution over time. The dots denote the observations while the line represents the ground truth obtained by numerical integration.

As in the previous publications, a squared exponential kernel was used. For FGPGM and MVGM, γ was set to 0.3, while AGM was provided with the true observation noise standard deviations σ. The standard deviation of the proposal distribution of FGPGM was chosen as 0.075 for state proposals and as 0.09 for parameter proposals to roughly achieve an acceptance rate of 0.234. For all algorithms, it was decided to use only one GP to fit both states. This effectively doubles the amount of observations and leads to more stable hyperparameter estimates. As the state dynamics are very similar, this approximation is feasible.

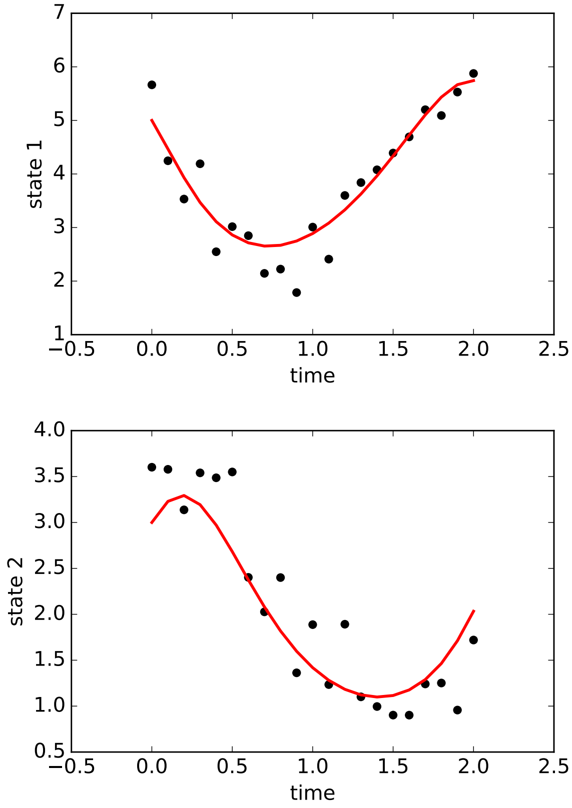


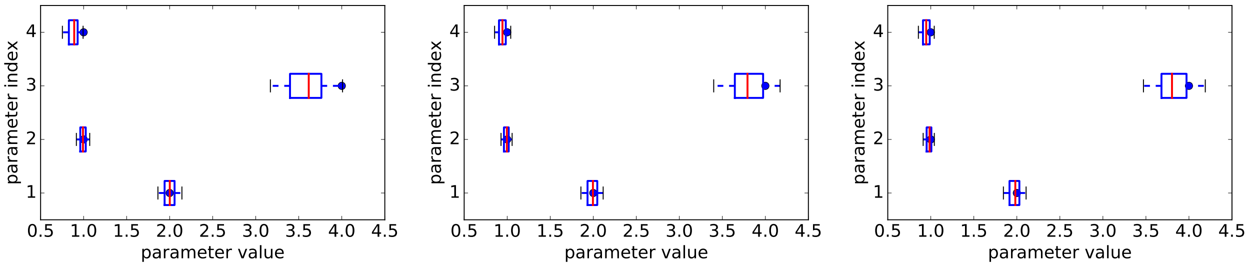
Figure 10: Example rollout of the Lotka Volterra system showing the state evolution over time for the high noise case. The dots denote the observations while the line represents the ground truth obtained by numerical integration.

7.7 Parameter Distribution

The MCMC approach of FGPGM allows to infer the probability distribution over parameters. This is shown for one example rollout in Figure 14. The inferred distributions are close to Gaussian in shape. This likely explains the sampling-like performance of the variational approach MVGM, as their assumptions of using a factorized Gaussian proxy distribution over the parameters seems to be a good fit for the true distribution.

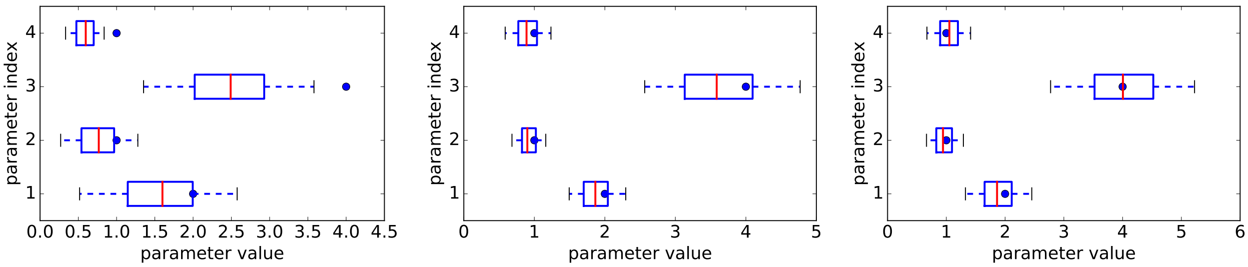
7.8 Protein Transduction

Figure 15 and Figure 16 show median plots for the states obtained by numerical integration of the inferred parameters of the Protein Transduction system.



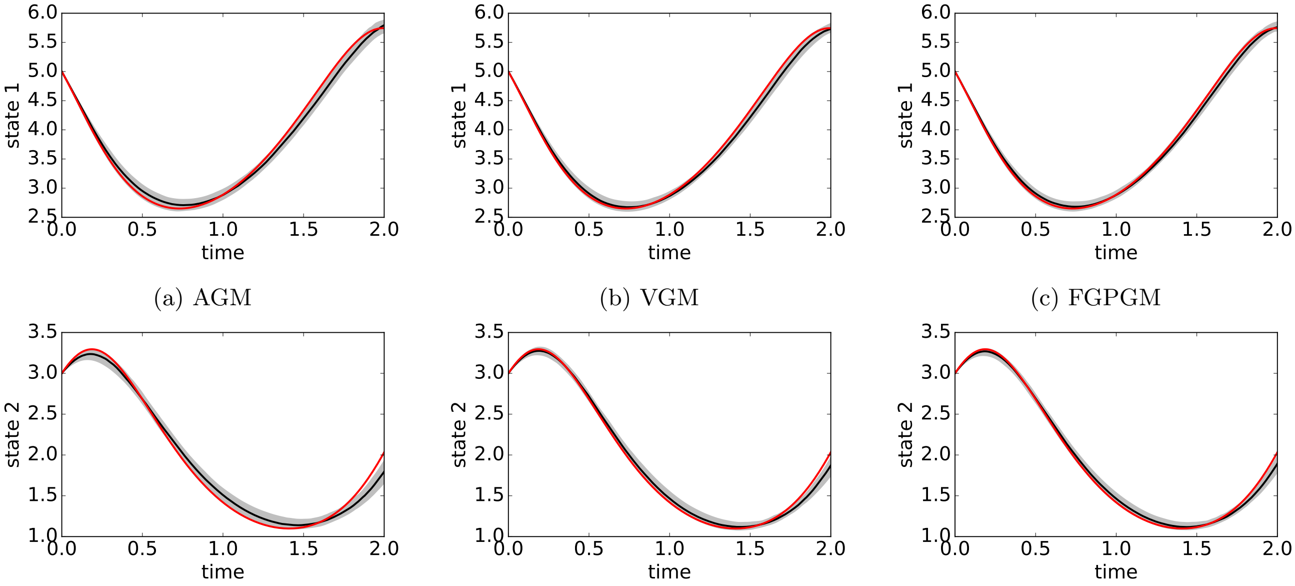
(a) AGM (b) FGPGM (c) MVGM

Figure 11: Inferred parameters for the low noise case. Ground truth (blue dots), median (red line), 50% (boxes) and 75% (whiskers) quantiles evaluated over 100 independent noise realizations are shown.



(a) AGM (b) FGPGM (c) MVGM

Figure 12: Boxplots showing the inferred parameters over 100 runs for the Lotka Volterra dynamics for the high noise case. The blue dots represent the ground truth, the red line denotes the median of the 100 parameter estimates while the boxes and whiskers denote 50% and 75% quantiles.



(a) AGM (b) VGM (c) FGPGM

Figure 13: Median Plots for all states of Lotka Volterra with low noise. State 1 is in the top row, state 2 is in the bottom row. The red line is the ground truth, while the black line and the shaded area denote the median and the 75% quantiles of the results of 100 independent noise realizations. As was already to be expected by the parameter estimates, FGPGM and VGM are almost indistinguishable while AGM falls off a little bit.

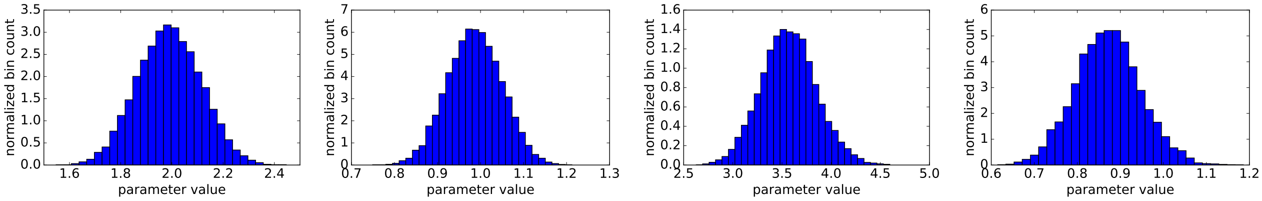


Figure 14: Histograms representing the MCMC samples obtained for one example run of the Lotka Volterra system. Each histogram represents the marginal distribution of one ODE parameter.

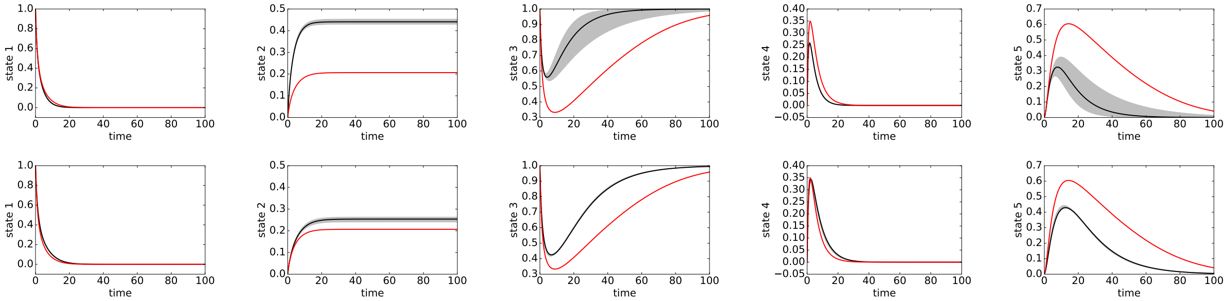


Figure 15: Median plots for all states of the most difficult benchmark system in the literature, Protein Transduction. The red line is the ground truth, while the black line and the shaded area denote the median and the 75% quantiles of the results of 100 independent noise realizations. FGPGM (middle) is clearly able to find more accurate parameter estimates than AGM (top).

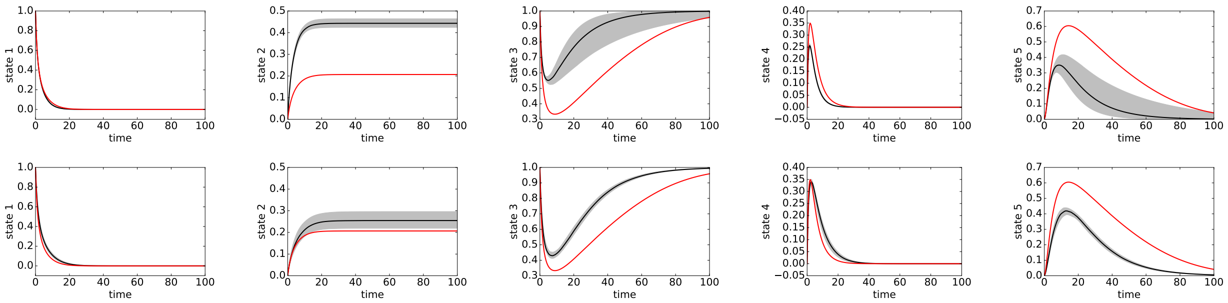
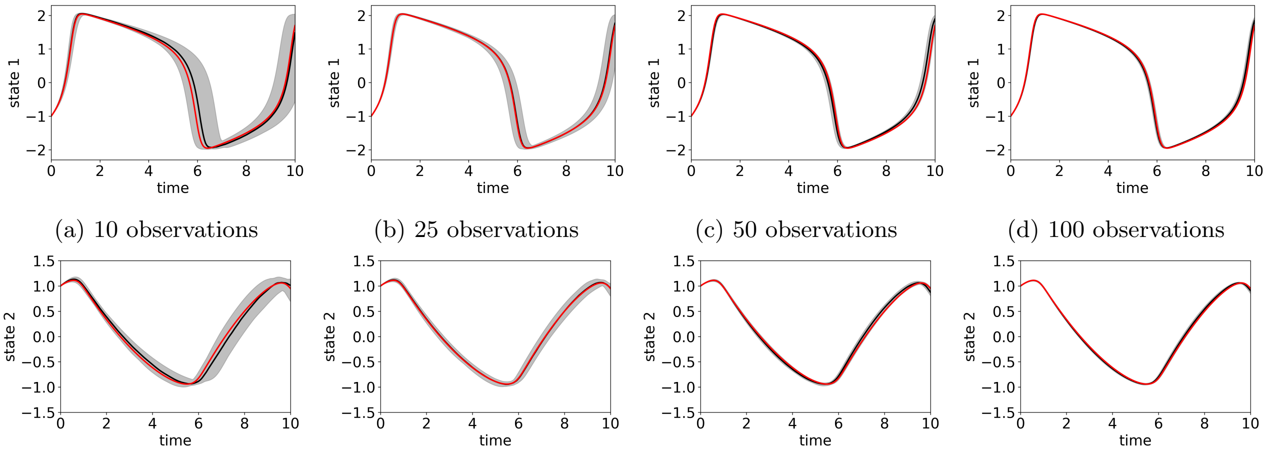


Figure 16: Median Plots for all states of the most difficult benchmark system in the literature, Protein Transduction, for the high noise case.The red line is the ground truth, while the black line and the shaded area denote the median and the 75% quantiles of the results of 100 independent noise realizations. FGPGM (middle) is clearly able to find more accurate parameter estimates than AGM (top).



(a) 10 observations (b) 25 observations (c) 50 observations (d) 100 observations

Figure 17: Median plots of the numerically integrated states after parameter inference for the FHN system with SNR 10. Ground truth (red), median (black) and 75% quantiles (gray) over 100 independent noise realizations.