Computational Statistics and Data Analysis 196 (2024) 107972

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Bayesian estimation of large-scale simulation models with Gaussian process regression surrogates

Sylvain Barde

School of Economics, University of Kent, Sibson Building, Park Wood Road, Canterbury CT2 7FS, UK

ARTICLE INFO

Keywords:
Linear model of coregionalization
Likelihood-free inference
Variational inference
Agent-based model
Non-parametric method

ABSTRACT

Large scale, computationally expensive simulation models pose a particular challenge when it comes to estimating their parameters from empirical data. Most simulation models do not possess closed-form expressions for their likelihood function, requiring the use of simulation-based inference, such as simulated method of moments, indirect inference, likelihood-free inference or approximate Bayesian computation. However, given the high computational requirements of large-scale models, it is often difficult to run these estimation methods, as they require more simulated runs that can feasibly be carried out. The aim is to address the problem by providing a full Bayesian estimation framework where the true but intractable likelihood function of the simulation model is replaced by one generated by a surrogate model trained on the limited simulated data. This is provided by a Linear Model of Coregionalization, where each latent variable is a sparse variational Gaussian process, chosen for its desirable convergence and consistency properties. The effectiveness of the approach is tested using both a simulated Bayesian computing analysis on a known data generating process, and an empirical application in which the free parameters of a computationally demanding agent-based model are estimated on US macroeconomic data.

1. Introduction

The increasing availability of computing power has led over time to simulation methods becoming part of the standard toolbox of researchers. Gilbert and Troitzsch (2005) argue that their appeal for the social sciences lie in their enabling a better understanding and formalization of the non-linearities or emergent phenomena pervasive in social structures. Conditional on the simulations being a valid representation of the social phenomenon of interest, this allows scenario analysis or simulated experiments to be carried out. However, establishing this precondition is challenging for social science simulations, particularly for agent-based models (ABMs), where aggregate properties of observable variables emerge from the bottom-up simulation of individual agent interactions (Fagiolo et al., 2007). Because ABMs form the hardest case of this validation problem, they will form the focus of the empirical application, however, the methodology proposed here is designed to be broadly applicable to any model producing simulated data with a time-series dimension.

Gouriéroux and Monfort (1993, 1996) provide an early overview of simulation-based inference methods, while Fagiolo et al. (2019) review how those approaches have been applied in the context of ABM estimation. Existing applications fall into two categories, moment-based and likelihood-based, both of which are special cases of the more general indirect inference framework

E-mail address: s.barde@kent.ac.uk.

https://doi.org/10.1016/j.csda.2024.107972
Received 21 August 2023; Received in revised form 17 April 2024; Accepted 17 April 2024

Available online 23 April 2024

(Gouriéroux et al., 1993; Smith, 1993). These differ in the metric chosen for the distance between empirical and simulated auxiliary parameters, with the Wald metric leading to moment-based methods, and the likelihood ratio metric to simulated likelihood methods (See Smith, 2016, on this point). The method proposed here is also a form of indirect inference, using GP regression as the auxiliary model. Historically, the first method implemented on ABMs is the method of simulated moments (MSM), in Gilli and Winker (2003), which was extended by the simulated minimum distance (SMD) method of Grazzini and Richiardi (2015). Likelihood-based methods involve simulated likelihood (Kukačka and Baruník, 2017) and Bayesian estimation approaches (Grazzini et al., 2017; Delli Gatti and Grazzini, 2020), with kernel density estimation (KDE) of the simulated data providing a non-parametric estimate of the likelihood.

Grazzini et al. (2017); Lamperti et al. (2018) make the point that these methods have so far been restricted to relatively simple ABMs, because the computational cost of running the simulations generates two problems when applied to large-scale models. First, the computational requirements of larger models impose a practical limit to the number of simulation runs that can be performed, assumed here to be on the order of 1000 runs, enough to run a Monte Carlo analysis and establish the statistical properties of a simulation. Second, their high parameter dimensionality, which usually underpins the computational requirements, also complicates the exploration of the parameter space through the dilution of the already limited compute budget in the high-dimensional sampling design, as well as the reduction in efficiency when using MCMC to sample from a high-dimensional posterior (Gelman et al., 1996, 1997). An additional issue, particularly acute for social science ABMs, is the compounding of this computational cost brought on by the frequent need to estimate models on multiple empirical datasets (e.g. for multiple countries, time-periods, etc.).

This paper proposes a Bayesian estimation framework with a Gaussian process regression surrogate (BEGRS) that is specifically designed to be tractable on computationally demanding models. It includes a demonstration on the Caiani et al. (2016) model, an ABM known for being computationally expensive. The strategy adopted is to proceed in two sequential stages. First, we generate a surrogate likelihood function for the model, based on Gaussian process (GP) regression, using the limited number of simulated runs to generate training data that spans the parameter space. In the second stage, this surrogate likelihood is used as part of a standard Bayesian framework to estimate the parameters of the simulation model from empirical data.

The use of a GP surrogate to reduce the computational burden of working with simulation models, especially ABMs, is not novel in itself: the seminal work of Kennedy and O'Hagan (2000, 2001) already focuses on reducing the computational expense of emulating expensive computer simulations with GP regression. A large and relatively recent literature on likelihood-free inference (LFI) also advocates the use of GPs as surrogates for models lacking a tractable likelihood, for example Meeds and Welling (2014), Gutmann and Corander (2016), Järvenpää et al. (2021) or Aushev et al. (2024), often based on the synthetic likelihood approach of Wood (2010). Similarly, within the economics ABM literature, Salle and Yıldızoğlu (2014), Bargigli et al. (2020) and Chen and Desiderio (2022) all provide examples of using GP regression as a surrogate for an ABM, typically using the GP to model simulated moments that can enter an MSM estimation.

The main divergence from these previous contributions is the fact that the GP surrogate used here does not rely on discrepancies between empirical and simulated data, which means that the training stage only requires simulated data, and the trained surrogate model can be reused 'as-is' when estimating on distinct empirical datasets. This design choice helps amortize the computational cost in settings where models require estimation on many datasets, at the cost of not being able use Bayesian optimization to actively acquire training parameter samples from regions with high empirical likelihood. Instead, we show that estimation is possible even when the entire training data is simulated in one preliminary step. Given this design choice, the efficiency improvements come instead from leveraging the time-series nature of the simulated data and including lagged values of the model variables in the GP regression inputs. This turns the surrogate into a one-step-ahead predictor, thus greatly increasing the effective amount of training data available. The second improvement is to rely on variational approximations to the GP, which allows the surrogate to scale more efficiently.

This proposed two-step approach is most closely related to several existing contributions. First is the KDE-based Bayesian estimation framework Grazzini et al. (2017) and Delli Gatti and Grazzini (2020). The key departure here lies in choosing a different non-parametric framework for generating the surrogate likelihood, in order to increase the efficiency of the methodology with respect to the number of simulations required. The second contribution is the recent work of Hooten et al. (2020), who use a GP surrogate to estimate a wildlife population ABM and an epidemiological ABM. The difference here lies both in the nature and scale of the ABMs involved, with ABMs in the social sciences being more highly parametrised and there being much more uncertainty over the deep rules that govern agent behaviour. Finally, this methodology is conceptually related to the literature using neural networks as the basis of surrogate models in LFI applications, for instance Lueckmann et al. (2019); Papamakarios et al. (2017, 2019), or even Platt (2021) in the context of ABM estimation. As will be discussed below, they are in a theoretical sense the closest to the methodology proposed here, due to the universal approximation property of neural networks that is shared with GPs.

The remainder of the paper is organized as follows. Section 2 details the variational GP regression framework used to generate the surrogate likelihood, section 3 presents the wider Bayesian estimation framework for simulation models. Two applications are provided in section 4, and section 5 concludes.

2. Gaussian process regression surrogate likelihood

2.1. One-step-ahead multivariate GP: setup and properties

The notation below follows the GP regression literature (specifically Hensman et al., 2015; Seeger et al., 2008; Burt et al., 2019) in order to facilitate presentation, but is adapted to deal both with the multivariate nature of the problem as well as the embedding of the GP regression into a wider Bayesian estimation framework. The former involves vectorizing over the observable variables and

adopting a block-diagonal structure for the covariance matrices. By casting the multivariate GP as a larger univariate process one can show that the properties of the quantities of interest in univariate GP regression extend to the multivariate case. To address the second problem, and limit the potential for confusion between the Bayesian estimation of the GP surrogate from the model simulation versus the wider Bayesian estimation of the simulation model itself, variables and parameters used in the latter estimation will be labelled with a superscript $*$. Tables 3–5 in appendix A provide a summary of the notation used.

There are M empirical observable variables, modelled with V latent GP variables. Where necessary, these are indexed respectively using a subscript m and v. There are $T > 1$ time series observations for each variable, indexed with subscript t. The simulated data is generated using S samples $\boldsymbol{\theta}_s$ drawn from a parameter space Θ. Lower case bold $\mathbf{x}, \mathbf{y}, \mathbf{u}$, etc. indicate vectors, assumed to be column vectors unless stated otherwise, while uppercase bold $\mathbf{A}, \mathbf{B}, \boldsymbol{\Sigma}$, etc. refer to matrices. $\mathbf{K}$ will specifically refer to the variance-covariance matrix produced by a kernel function, and superscripts attached to kernel matrixes, e.g. $\mathbf{K}^{\mathbf{x},\mathbf{x}}$, will refer to the inputs of the kernel function. Braces are used to refer to the full set of vectors or matrices attached to observable and latent variables, e.g. $\{\mathbf{A}_v\} = \{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_V\}$ is the set of all $\mathbf{A}_v$ matrices attached to the latent variables. Two vectorization operations are required to convert these sets into single objects.

$$\text{vec}(\{\mathbf{f}_v\}) = \begin{bmatrix} \mathbf{f}_1^T & \mathbf{f}_2^T & \dots & \mathbf{f}_V^T \end{bmatrix}^T, \qquad \text{bdiag}(\{\mathbf{A}_v\}) = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_V \end{bmatrix} \tag{1}$$

The first is the standard vectorization operation which stacks multiple column vectors into a single vector, the second constructs a block diagonal matrix from a set of matrices, with $\mathbf{0}$ being an appropriately sized null matrix.

Let $\mathbf{Y}^*$ be a $T \times M$ matrix of empirical data and let $\mathbf{Y} = \{\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_S\}$ be a set of S simulated $T \times M$ real-valued matrices, each obtained by simulating the model with a row vector of d_Θ parameters $\boldsymbol{\theta}_s \in \Theta$. It is assumed w.l.o.g. that both the simulated and empirical data have T time-series observations. The values in the columns of $\mathbf{Y}_s$ and $\mathbf{Y}^*$ are assumed to be centred. Let $\boldsymbol{\theta}^*$ be a vector of simulation model parameters to be estimated from the empirical data $\mathbf{Y}^*$ using Bayesian methods. Then, ignoring the marginal data density $p(\mathbf{Y}^*)$, the object of interest is the posterior density of $\boldsymbol{\theta}^*$ given $\mathbf{Y}^*$, i.e. $p(\boldsymbol{\theta}^* \mid \mathbf{Y}^*) \propto p(\mathbf{Y}^* \mid \boldsymbol{\theta}^*) p(\boldsymbol{\theta}^*)$. The prior probability $p(\boldsymbol{\theta}^*)$ is known in advance, so the likelihood $p(\mathbf{Y}^* \mid \boldsymbol{\theta}^*)$ is the term of interest. One can take advantage of the time-series structure of the data to decompose the log-likelihood into a sum of individual one-step-ahead contributions for each time-series observation in the dataset:

$$\ln p(\mathbf{Y}^* \mid \boldsymbol{\theta}^*) = \sum_t \ln p(\mathbf{Y}_t^* \mid \boldsymbol{\Omega}_{t-1}^*, \boldsymbol{\theta}^*) \tag{2}$$

Where $\boldsymbol{\Omega}_t^* = \{\mathbf{Y}_t^*, \mathbf{Y}_{t-1}^*, \dots \mathbf{Y}_1^*\}$ is the information set at t. The key challenge for simulation models is the lack of a closed-form expression for the predictive density $p(\mathbf{Y}_t^* \mid \boldsymbol{\Omega}_{t-1}^*, \boldsymbol{\theta}^*)$. The proposed methodology uses instead an approximation to the true likelihood (2) produced by a surrogate model, where as a first simplification $\boldsymbol{\Omega}_t^* = \mathbf{Y}_t$, i.e. we approximate the true process as a first-order Markov process:

$$\ln p(\mathbf{Y}^* \mid \boldsymbol{\theta}^*) \approx \ln \hat{p}(\mathbf{Y}^* \mid \boldsymbol{\theta}^*) = \sum_t \ln p(\mathbf{Y}_t^* \mid \mathbf{Y}_{t-1}^*, \boldsymbol{\theta}^*, \mathbf{Y}, \boldsymbol{\theta}, \boldsymbol{\phi}) \tag{3}$$

Where $p(\mathbf{Y}_t^* \mid \mathbf{Y}_{t-1}^*, \boldsymbol{\theta}^*, \mathbf{Y}, \boldsymbol{\theta}, \boldsymbol{\phi})$ is the one-step-ahead density for observation $\mathbf{Y}_t^*$, conditional on $\mathbf{Y}_{t-1}^*$ and $\boldsymbol{\theta}^*$, provided by a surrogate model with internal parameters $\boldsymbol{\phi}$, optimized on a simulated training set $\mathbf{Y}$ generated from parameter samples $\boldsymbol{\theta}$. This is known in the literature as a Vecchia approximation, and explains why the methodology requires $T > 1$, to ensure at least one Markov transition is available. In principle, more lags can be included to generate a higher-order Markov approximation, at the cost of a higher dimensionality of the input space.

Two notational clarifications are needed. First, because GPs are always conditioned on $\mathbf{Y}, \boldsymbol{\theta}$ and $\boldsymbol{\phi}$, in line with the GP regression literature, explicit mention of these conditioning variables is dropped and $\hat{p}$ is used to indicate the use of a GP surrogate. Second, again to ensure consistency with the GP literature, the conditioning variables $\mathbf{Y}, \mathbf{Y}^*, \boldsymbol{\theta}, \boldsymbol{\theta}^*$ are packaged into two sets of inputs $\mathbf{X}, \mathbf{X}^*$. These are given by the following block matrices, with $\mathbf{1}_{T-1}$, a $T-1$ length column vector of ones, $\mathbf{Y}_s$, the simulated observations generated by the model using parameter setting $\boldsymbol{\theta}_s$, and $\mathbf{L}$, the first-order lag operator:

$$\mathbf{X} = \begin{bmatrix} \mathbf{L}\mathbf{Y}_1 & \mathbf{1}_{T-1}\boldsymbol{\theta}_1 \\ \mathbf{L}\mathbf{Y}_2 & \mathbf{1}_{T-1}\boldsymbol{\theta}_2 \\ \vdots & \vdots \\ \mathbf{L}\mathbf{Y}_S & \mathbf{1}_{T-1}\boldsymbol{\theta}_S \end{bmatrix}, \qquad \mathbf{X}^* = \begin{bmatrix} \mathbf{L}\mathbf{Y}^* & \mathbf{1}_{T-1}\boldsymbol{\theta}^* \end{bmatrix} \tag{4}$$

$\mathbf{X}$ is an $N \times d$ matrix of training inputs, with $N = S(T-1)$ training observations and $d = M + d_\Theta$ dimensions. In the terminology of Kennedy and O'Hagan (2001), the GP inputs $\mathbf{X}$ consist of *calibration inputs* $\boldsymbol{\theta}_s$ and *variable inputs* $\mathbf{L}\mathbf{Y}_s$. This maximizes the amount of training data available for the GP surrogate from the S simulation runs by treating each individual observation as a training data point, barring those lost by taking a time lag. Where necessary, $x_i, x_j, \dots$ and $x_i^*, x_j^*, \dots$ and will denote individual rows of $\mathbf{X}$ and $\mathbf{X}^*$ respectively, i.e. individual observations, and $\mathcal{X}$ denotes the d-dimensional input space from which x_i and x_i^* are drawn. It is also convenient to define $\mathbf{y} = \text{vec}(\mathbf{Y} \setminus y_1)$ and $\mathbf{y}^* = \text{vec}(\mathbf{Y}^* \setminus y_1^*)$ as the vectorizations of the simulated and empirical data, with the first row removed to allow for the time lag.

The multivariate nature of the surrogate is handled using a Linear Model of Coregionalization (LMC) (Alvarez et al., 2012), which builds on the multi-output emulation of Conti and O'Hagan (2010). The LMC generates predictions for M observable variables based on V latent variables, each modelled by a standard univariate GP. In general, assuming a set of arbitrary Gaussian distributions $\mathbf{f}_v \sim \mathcal{N}(\boldsymbol{\mu}_v, \boldsymbol{\Psi}_v)$ for the V latent predictions and a $M \times V$ weights matrix $\mathbf{B}$, the distribution of the LMC prediction is $\tilde{\mathbf{f}} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Psi}})$, with the mean and variance given by the following expressions:

$$\begin{cases} \tilde{\boldsymbol{\mu}} = \tilde{\mathbf{B}}\boldsymbol{\mu} \\ \tilde{\boldsymbol{\Psi}} = \tilde{\mathbf{B}}\boldsymbol{\Psi}\tilde{\mathbf{B}}^T = \sum_v \left(\mathbf{B}_v \mathbf{B}_v^T \otimes \boldsymbol{\Psi}_v\right) \end{cases} \tag{5}$$

Where $\boldsymbol{\mu} = \text{vec}(\{\boldsymbol{\mu}_v\})$, $\boldsymbol{\Psi} = \text{bdiag}\left(\{\boldsymbol{\Psi}_v\}\right)$, and $\tilde{\mathbf{B}} = \mathbf{B} \otimes \mathbf{I}_N$. The second equality for $\tilde{\boldsymbol{\Psi}}$ corresponds to the more traditional LMC representation of Alvarez et al. (2012). The measurement error on each observable m is assumed to be i.i.d. Gaussian with standard deviation σ_m and uncorrelated across variables, so that the variance of the prediction error below is $\boldsymbol{\Sigma}^2 = \text{bdiag}\left(\{\sigma_m^2 \mathbf{I}_N\}\right)$.

$$p(\mathbf{y} \mid \tilde{\mathbf{f}}) \;=\; \mathcal{N}(\mathbf{y} \mid \tilde{\boldsymbol{\mu}}\,,\, \boldsymbol{\Sigma}^2) \tag{6}$$

The prediction components $\mathbf{f}_v$ of the latent variables used in the proposed methodology are each assumed to follow a zero-mean Gaussian distribution, $\mathbf{f}_v \sim \mathcal{N}(0, \mathbf{K}_v^{\mathbf{x},\mathbf{x}})$ where covariances $[\mathbf{K}_v^{\mathbf{x},\mathbf{x}}]_{i,j} = K_v(x_i, x_j)$ are modelled using a kernel function $K_v(.,.)$. Specifically, we use the following radial basis function (RBF) kernel, where ℓ_v is the variable-specific kernel length scale:

$$K_v(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\ell_v^2}\right) \tag{7}$$

The LMC increases the flexibility of the model by having V distinct kernels, each capturing correlations in the training data at different length scales ℓ_v. The set of length scales $\{\ell_v\}$ will play an important role in the convergence properties of the GP surrogate, discussed below. In addition, because typically $V < M$, the LMC also reduces the dimensionality of the surrogate model. Given (5) and with $\mathbf{K}^{\mathbf{x},\mathbf{x}} = \text{bdiag}\left(\{\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\}\right)$, the distribution of the LMC predictions on the training data $\mathbf{X}$ is $\tilde{\mathbf{f}} \sim \mathcal{N}(0, \tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}})$, with variance $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} = \tilde{\mathbf{B}}\mathbf{K}^{\mathbf{x},\mathbf{x}}\tilde{\mathbf{B}}^T$. Marginalizing out the vectorized GP latents $\mathbf{f}$ results in the following likelihood of the evidence for the LMC model:

$$\hat{p}(\mathbf{y}) = \int p(\mathbf{y} \mid \tilde{\mathbf{f}}) p(\tilde{\mathbf{f}}) d\mathbf{f} = \mathcal{N}\left(\mathbf{y} \mid 0, \tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2\right) \tag{8}$$

The log-likelihood of the evidence is a function of the underlying parameters $\boldsymbol{\phi} = \{\mathbf{B}, \ell_v, \sigma_m\}$, which implies that optimal values for $\boldsymbol{\phi}$ can be obtained by maximization.

$$\ln \hat{p}(\mathbf{y}) = -\frac{NM}{2}\ln(2\pi) - \frac{1}{2}\mathbf{y}^T\left(\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2\right)^{-1}\mathbf{y} - \frac{1}{2}\ln|\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2| \tag{9}$$

Once the optimal GP parameters $\boldsymbol{\phi}$ have been obtained from the training data, the LMC surrogate provides multivariate Gaussian predictions $p(\tilde{\mathbf{f}}^* \mid \mathbf{X}^*) = \mathcal{N}\left(\tilde{\mathbf{f}}^* \mid \tilde{\boldsymbol{\mu}}^*, \tilde{\boldsymbol{\Psi}}^*\right)$ for previously unseen inputs $\mathbf{X}^*$, with prediction mean and variance functions given by:

$$\begin{cases} \tilde{\boldsymbol{\mu}}^* = \tilde{\mathbf{K}}^{\mathbf{x}^*,\mathbf{x}}\left(\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2\right)^{-1}\mathbf{y} \\ \tilde{\boldsymbol{\Psi}}^* = \tilde{\mathbf{K}}^{\mathbf{x}^*,\mathbf{x}^*} - \tilde{\mathbf{K}}^{\mathbf{x}^*,\mathbf{x}}\left(\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2\right)^{-1}\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}^*} \end{cases} \tag{10}$$

2.2. Theoretical properties of the multivariate GP surrogate

The use of GP regression for the surrogate model is motivated by several desirable theoretical properties, the first being universal approximation. Let $\mathcal{H}$ be the reproducing kernel Hilbert space (RKHS) of all functions f defined as linear combinations of the eigenfunctions of kernel (7), let $\mathcal{Z}$ be a compact subset of $\mathcal{X}$, and let $C(\mathcal{Z})$ be the space of continuous functions over $\mathcal{Z}$. Given $f_0 \in C(\mathcal{Z})$ Micchelli et al. (2006) shows that if the kernel is universal, then there exists a function $f \in \mathcal{H}$ that approximates f_0 arbitrarily well, i.e. $\|\mathbf{f} - \mathbf{f}_0\|_\infty < \varepsilon$. In such cases, GP regression possesses the universal approximation property and $\mathcal{H} = C(\mathcal{Z})$. For the specific methodology proposed here: theorem 17 of Micchelli et al. (2006) proves that the RBF kernel (7) is universal, and theorem 12 of Caponnetto et al. (2008) proves that a multivariate kernel of the form $\mathbf{M} \times K(x_i, x_j)$ is universal if and only if K is itself universal and $\mathbf{M}$ is positive definite. This is the case for the LMC model (5) if $V = M$, as $\mathbf{M} = \mathbf{B}_v\mathbf{B}_v^T$ is positive definite by construction. If $V < M$, then $\mathbf{B}_v\mathbf{B}_v^T$ is only positive semidefinite, however a Tikhonov regularization $\mathbf{M} = \mathbf{B}_v\mathbf{B}_v^T + a\mathbf{I}$ can be performed, for small a. This procedure, known as 'adding jitter', ensures universality of $\mathbf{M} \times K(x_i, x_j)$, but can change the GP estimates of the (regularized) $\mathbf{B}_v$. Finally, Wynne et al. (2021) show that a closed bounded interval $\mathcal{Z} \subset \mathbb{R}^d$ meets the conditions required for GP regression to be consistent. As a result, the LMC kernel (5) possesses the universal approximation property on a bounded subset of $\mathbb{R}^d$.

The second property relates to the rate at which the mean GP prediction (10) converges to the true function f_0 as the size of training data set $\mathbf{X}$ increases. A wide range of results in the literature establishes asymptotic convergence in the univariate setting, divided between work that uses an integrated mean square error (IMSE) loss, such as Choi and Schervish (2007); Shi and Choi (2011); Le Gratiet and Garnier (2015); Koepernik and Pfaff (2021), and works that use a Kullback-Leibler (KL) loss, such as Seeger et al. (2008); Burt et al. (2019). The two approaches can in fact be reconciled, see Van Der Vaart and Van Zanten (2011). Le Gratiet and Garnier (2015) in particular prove almost sure convergence for non-degenerate Mercer kernels with bounded features, such as (7).

However, given the computationally constrained setting chosen here, the bigger concern is the rate of this convergence. Seeger et al. (2008) argue that in such a setting, the concept of interest is information consistency, where one examines the asymptotic behaviour of the expected KL divergence from the surrogate to the noise distribution (6). This can be bounded using the standard formula for the KL divergence between two Gaussians:

$$D_{KL}\left[p(\mathbf{y} \mid \tilde{\mathbf{f}}) \parallel \hat{p}(\mathbf{y})\right] \quad \leq \quad \frac{1}{2}\tilde{\boldsymbol{\mu}}^T \left(\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2\right)^{-1} \tilde{\boldsymbol{\mu}} + \frac{1}{2} \ln \frac{|\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2|}{|\boldsymbol{\Sigma}^2|} \tag{11}$$

The mean predictions $\tilde{\boldsymbol{\mu}}$ are in the RKHS $\tilde{\mathcal{H}}$ of the LMC kernel corresponding to $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} + \boldsymbol{\Sigma}^2$ so the first term is simply the Hilbert norm of the prediction $\|\tilde{\boldsymbol{\mu}}\|^2_{\tilde{\mathcal{H}}}$. The second term, known as the regret, is equal up to an additive constant to the last term in the log-likelihood (9):

$$R = \frac{1}{2} \ln \left| \mathbf{I}_{MN} + \boldsymbol{\Sigma}^{-2} \tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} \right| \tag{12}$$

With $\nu(x)$ the distribution of the input data, GP prediction is informationally consistent if $N^{-1} E_{\nu(x)}\left[D_{KL}\left[p(\mathbf{y} \mid \tilde{\mathbf{f}}) \parallel \hat{p}(\mathbf{y})\right]\right] \to 0$ as $N \to \infty$, i.e. the expected KL divergence (11) per training draw from $\mathcal{X}$ goes to zero as the size of the training set increases. Crucially, because $\|\tilde{\boldsymbol{\mu}}\|^2_{\tilde{\mathcal{H}}} < \infty$, the first term vanishes asymptotically and the scaling behaviour of the regret term (12) alone determines the information consistency. Several results have established an upper bound on $E_{\nu(x)}[R]$ of $\mathcal{O}\left((\log N)^{d+1}\right)$ for univariate GP regression with a d-dimensional training space $\mathcal{X}$, including Seeger et al. (2008). Proposition 1, below, shows that this bound extends to the LMC setting.

Proposition 1. *The expected regret of the LMC* (12) *has the following upper bound, where* υ^* *indicates the latent GP variable possessing the largest regret:*

$$E_{\nu(x)}\left[\ln \left| \mathbf{I}_{MN} + \boldsymbol{\Sigma}^{-2} \tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} \right| \right] \quad < \quad MV \sum_{h=0}^{\infty} \ln \left(1 + V b^2_{\upsilon^*,m^*} \sigma^{-2}_{m^*} \lambda_{\upsilon^*,h} N\right)$$

Proof. This is provided in appendix B.

Corollary. *Let the observations* x *be distributed with density* $\nu(x) = \mathcal{N}(0, 4a^2\mathbf{I}_d)$ *for a constant* a*. Then the expected regret of the LMC* (12) *with RBF kernels* (7) *on* $\mathbf{X}$ *is* $\mathcal{O}\left((\log N)^{d+1}\right)$.

This is immediate from the fact that the bound in Proposition 1 is the same, up to some multiplicative constants, as the one in Seeger et al. (2008) for the case of a single RBF kernel function. Assuming the same distribution as theirs for the inputs, the LMC regret with V RBF kernels will therefore scale with N at the same rate as that of a single RBF kernel. Note, however, that the regret of the LMC kernel (12) will be larger than the single kernel case of Seeger et al. (2008) due to the fact that V copies of the worst-performing kernel enter the regret term, which itself is multiplied by MV.

A third desirable property of GP regression in the context of surrogate modelling is the auto-regularized nature of the log-likelihood (9) maximization, where the minimization of the squared deviation of the GP prediction from the data is penalized by the regret (12), enforcing smoothness of the GP and limiting the risk of overfitting the training data. In fact, Bishop (2006, sections 3.1.4 and 6.4) shows that for linear kernels $K(x_i, x_j) = x_i^T x_j$, GP regression is equivalent to ridge regression. This enables a comparison with the neural network approaches mentioned in the introduction, which also possess the universal approximation property, as proven by Hornik et al. (1989). Neal (1996) establishes that assuming a zero-mean Gaussian prior for the network parameters, a single hidden layer neural network will converge under Bayesian learning to a GP as the number of hidden units goes to infinity (see also Bishop, 2006; Rasmussen and Williams, 2006; Burt et al., 2019). This is functionally equivalent to the RKHS representation of GP regression, where any smooth function can be approximated arbitrarily well by an infinite weighted sum over kernel eigenfunctions. Neal (1996) points out that the width of a neural network is a hyperparameter that needs to be optimized over network designs, in order to avoid over-fitting the training data, unlike GP regression. What GPs lose in this trade-off is the extra flexibility available from using multiple hidden layers in a deep network, for instance being able to model discontinuities, which will impose small length scales ℓ_υ and high regret (12). Given the computationally constrained setting investigated here, however, the design choice is to prefer a simpler, auto-regularizing model that generalizes well from an ex-ante fixed amount of training data.

Two additional properties are worth mentioning. First, the two-step design of the methodology minimizes the computationally expensive runs of the simulation model when estimating the model on multiple empirical datasets. Because the GP's training only relies on simulated data, and not empirical data, the same surrogate can be used when estimating model parameters on different empirical datasets, thus amortizing the computationally expensive training data. Second, the analytical tractability of the Gaussian predictions ensures differentiability of the surrogate likelihood. This specific aspect facilitates maximization of, and sampling from, the posterior through the use of gradient-based methods.

Finally, two caveats on the universality property and convergence need to be mentioned. First, the universality property only ensures that the one-step-ahead LMC predictions converge to the first-order Markov process assumed in the Vecchia approximation (3), and offers no guarantee that this approximation to the full likelihood (2) is valid in the first place. Second, a key practical implication of Proposition 1 is that the convergence of the LMC will tend to be determined by the regret of the kernel associated with

the smallest length scale ℓ_{v^*}. Intuitively, if the output variable $\mathbf{y}$ changes rapidly over small ranges of $\mathcal{X}$ (for example because of a discontinuity), a small length scale will be required to accurately capture this, and for a given amount of training data N randomly spread over $\mathcal{X}$, the prediction error will be larger than that of a smoother model, where the inputs can be captured with larger length scales. The practical implications of both issues are discussed further in section 3.1.

2.3. Scaling up: a variational approximation to the LMC

Obtaining exact GP predictions (10) for large N is not feasible in practice, due to the $\mathcal{O}(N^3)$ matrix inversions required. Several approaches aim at improving on this bound, such as the Nyström sampling approach of Rudi et al. (2015); Lu et al. (2020), or the sparse variational approach of Titsias (2009) and Hensman et al. (2015). Both take advantage of redundant information in the training data, reducing the large number of training points to a smaller set of sufficient statistics that retain critical information, thus improving computational efficiency. Following the suggestion of Gutmann and Corander (2016), we use the latter of the two here, and a small number of additional inducing points are introduced to augment the training observations. In terms of notation, these inducing points correspond to a set of inducing locations $\{\mathbf{Z}_v\}$ which generate inducing values $\{\mathbf{u}_v\}$, both of which are additional parameters of the GP that need to be estimated during the training stage. The full derivation of the variational approximation to the LMC is provided in online appendix A.

The joint distribution of the training predictions $\mathbf{f}_v$ and inducing values $\mathbf{u}_v$ associated with a given latent variable is now given by the following multivariate Gaussian:

$$p(\mathbf{f}_v, \mathbf{u}_v) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f}_v \\ \mathbf{u}_v \end{bmatrix} \middle| \, 0, \begin{bmatrix} \mathbf{K}_v^{\mathbf{x},\mathbf{x}} & \mathbf{K}_v^{\mathbf{x},\mathbf{z}} \\ \mathbf{K}_v^{\mathbf{z},\mathbf{x}} & \mathbf{K}_v^{\mathbf{z},\mathbf{z}} \end{bmatrix} \right) \tag{13}$$

The block structure of (13) means that Shur's complement can be used to obtain the latent predictions $\mathbf{f}_v$ conditional on the inducing values $\mathbf{u}_v$:

$$p(\mathbf{f}_v \mid \mathbf{u}_v) = \mathcal{N}(\mathbf{f}_v \mid \mathbf{A}_v \mathbf{u}_v, \, \mathbf{K}_v^{\mathbf{x},\mathbf{x}} - \mathbf{Q}_v^{\mathbf{x},\mathbf{x}}) \tag{14}$$

With:

$$\begin{cases} \mathbf{A}_v = \mathbf{K}_v^{\mathbf{x},\mathbf{z}} \left(\mathbf{K}_v^{\mathbf{z},\mathbf{z}}\right)^{-1} \\ \mathbf{Q}_v^{\mathbf{x},\mathbf{x}} = \mathbf{K}_v^{\mathbf{x},\mathbf{z}} \left(\mathbf{K}_v^{\mathbf{z},\mathbf{z}}\right)^{-1} \mathbf{K}_v^{\mathbf{z},\mathbf{x}} \end{cases} \tag{15}$$

The key computational gain is that obtaining (14) now only requires inverting $\mathbf{K}_v^{\mathbf{z},\mathbf{z}}$, which by design is much smaller than $\mathbf{K}_v^{\mathbf{x},\mathbf{x}}$. The marginal distribution of the inducing values is simply:

$$p(\mathbf{u}_v) = \mathcal{N}(\mathbf{u}_v \mid 0, \mathbf{K}_v^{\mathbf{z},\mathbf{z}}) \tag{16}$$

The conditional (14) and prior distributions (16) are assumed to be independent across latents v, which means that the equivalent distributions for the vectorized values $\mathbf{f} = \text{vec}(\{\mathbf{f}_v\})$ and $\mathbf{u} = \text{vec}(\{\mathbf{u}_v\})$ can be expressed as follows, with $\mathbf{A} = \text{bdiag}\left(\{\mathbf{A}_v\}\right)$, $\mathbf{K}^{\mathbf{x},\mathbf{x}} = \text{bdiag}\left(\{\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\}\right)$ and $\mathbf{Q}^{\mathbf{x},\mathbf{x}} = \text{bdiag}\left(\{\mathbf{Q}_v^{\mathbf{x},\mathbf{x}}\}\right)$:

$$\begin{cases} p(\mathbf{f} \mid \mathbf{u}) = \mathcal{N}(\mathbf{f} \mid \mathbf{A}\mathbf{u}, \, \mathbf{K}^{\mathbf{x},\mathbf{x}} - \mathbf{Q}^{\mathbf{x},\mathbf{x}}) \\ p(\mathbf{u}) = \mathcal{N}(\mathbf{u} \mid 0, \mathbf{K}^{\mathbf{z},\mathbf{z}}) \end{cases} \tag{17}$$

The conditional distribution $p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})$ is approximated by the following variational distribution $q(\mathbf{f}, \mathbf{u})$, obtained by combining (14) with a variational prior on $\mathbf{u}$, with mean $\mathbf{m} = \text{vec}(\{\mathbf{m}_v\})$ and covariance $\mathbf{S} = \text{bdiag}(\{\mathbf{S}_v\})$:

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f} \mid \mathbf{u}) q(\mathbf{u}), \qquad q(\mathbf{u}) = \mathcal{N}(\mathbf{u} \mid \mathbf{m}, \mathbf{S}) \tag{18}$$

Formally, the variational approach aims to minimize the KL divergence from $q(\mathbf{f}, \mathbf{u})$ to $p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})$. When rearranged, this expression provides the evidence lower bound (ELBO) for the model, which is a function of the variational GP's parameters $\boldsymbol{\phi} = \{\mathbf{Z}, \mathbf{m}, \mathbf{S}, \mathbf{B}, \boldsymbol{\ell}, \boldsymbol{\sigma}\}$.

$$\mathcal{L}(\boldsymbol{\phi}) = \int_{\mathbf{u}} \int_{\mathbf{f}} \ln \frac{p(\mathbf{f}, \mathbf{u}, \mathbf{y})}{q(\mathbf{f}, \mathbf{u})} q(\mathbf{f}, \mathbf{u}) \, d\mathbf{f} \, d\mathbf{u} = \ln p(\mathbf{y}) - D_{KL}\left[q(\mathbf{f}, \mathbf{u}) \parallel p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})\right] \tag{19}$$

Given that $\ln p(\mathbf{y})$ does not depend on $\boldsymbol{\phi}$, maximizing the ELBO (19) is equivalent to minimizing the KL divergence term. In addition, assuming that the variational distribution $q(\mathbf{f}, \mathbf{u})$ successfully approximates $p(\mathbf{f}, \mathbf{u} \mid \mathbf{y})$, the KL divergence will tend to zero, and maximizing $\mathcal{L}(\boldsymbol{\phi})$ becomes equivalent to maximizing the exact GP likelihood (9). A more tractable expression for $\mathcal{L}(\boldsymbol{\phi})$ can be obtained by evaluating the integral in (19).

$$\mathcal{L}(\boldsymbol{\phi}) = \ln \mathcal{N}(\mathbf{y} \mid \tilde{\mathbf{B}}\mathbf{A}\mathbf{m}, \boldsymbol{\Sigma}^2) - \frac{1}{2} \text{tr}\left(\boldsymbol{\Sigma}^{-1} \tilde{\mathbf{B}} \boldsymbol{\Psi} \tilde{\mathbf{B}}^T \boldsymbol{\Sigma}^{-1}\right) - D_{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u})\right] \tag{20}$$

Where $\boldsymbol{\Psi} = \mathbf{A}\mathbf{S}\mathbf{A}^T + \mathbf{K}^{\mathbf{x},\mathbf{x}} - \mathbf{Q}^{\mathbf{x},\mathbf{x}}$ is the variational variance-covariance matrix of the vectorized latents $\mathbf{f}$.

As in Hensman et al. (2015), the ELBO $\mathcal{L}(\boldsymbol{\phi})$ is optimized with stochastic gradient descent on random sub-samples of the training data $\{\mathbf{X}, \mathbf{y}\}$, further improving tractability. The consistency and convergence of variational approximations is proven by Burt et al.

(2019, 2020) by extending the analysis of Seeger et al. (2008). As long as the number of inducing variables scales as $\mathcal{O}\left((\log N)^d\right)$, then for RBF kernels $N^{-1}D_{KL}\left[q(\mathbf{u}) \parallel p(\mathbf{u})\right] \to 0$, and $E\left[\mathcal{L}(\phi)\right] \to E\left[\ln p(\mathbf{y})\right]$, which implies that the predictions of the variational GP converge to those of a full GP.

Once the LMC is trained on the simulated data, predictions for unseen input $\mathbf{X}^*$ are given by the variational distribution of $\mathbf{f}^*$, obtained by marginalizing the inducing values $\mathbf{u}$ out of the joint variational distribution (18). Given the first-order Markov assumption in surrogate likelihood (3) and the fact that the predictions are already conditioned on lagged observations (4), the contribution of each $\left(\mathbf{y}_{t-1}, \mathbf{y}_t\right)$ transition should enter the likelihood independently. As a result, only the main diagonal of the variance-covariance matrix $\mathbf{\Psi}$ is required to obtain the variational density of the LMC predictions $\tilde{\mathbf{f}}^*$.

$$q(\tilde{\mathbf{f}}^*) = \mathcal{N}\left(\tilde{\mathbf{f}}^* \mid \tilde{\boldsymbol{\mu}}, \tilde{\mathbf{\Psi}}\right) \tag{21}$$

Where given $\text{diag}(\mathbf{\Psi})$, a diagonal matrix containing the main diagonal of $\mathbf{\Psi}$, the mean and variance-covariance are:

$$\begin{cases} \tilde{\boldsymbol{\mu}} = \tilde{\mathbf{B}}\mathbf{A}\mathbf{m} \\ \tilde{\mathbf{\Psi}} = \tilde{\mathbf{B}} \times \text{diag}(\mathbf{\Psi})\tilde{\mathbf{B}}^T \end{cases} \tag{22}$$

Given (21), the surrogate likelihood of the empirical $\mathbf{y}^*$ data using the LMC is:

$$\hat{p}(\mathbf{y}^* \mid \boldsymbol{\theta}^*) = \int p(\mathbf{y}^* \mid \tilde{\mathbf{f}}^*) q(\tilde{\mathbf{f}}^*) d\mathbf{f}^* = \mathcal{N}\left(\mathbf{y}^* \mid \tilde{\boldsymbol{\mu}}, \tilde{\mathbf{\Psi}} + \mathbf{\Sigma}^2\right) \tag{23}$$

It is straightforward to derive the gradient of the surrogate likelihood with respect to the underlying simulation model parameters $\boldsymbol{\theta}^*$. Because these form part of the inputs $\mathbf{X}^*$ used in the GP prediction, they only enter the surrogate likelihood (23) through the RBF kernel (7) used to compute $\mathbf{K}_v^{\mathbf{x}^*,\mathbf{z}}$.

$$\frac{\partial \mathbf{K}_v^{\mathbf{x}^*,\mathbf{z}}}{\partial \boldsymbol{\theta}_i^*} = \frac{\left(\mathbf{Z}_i - \boldsymbol{\theta}_i^*\right)\left(\mathbf{1}_d\right)^T}{\ell_v^2} \odot \mathbf{K}_v^{\mathbf{x}^*,\mathbf{z}} \tag{24}$$

The gradient of the surrogate likelihood (23) can be obtained by applying the chain rule on the standard derivative of a Gaussian likelihood:

$$\begin{aligned} \frac{\partial \ln \hat{p}(\mathbf{y} \mid \boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}_i^*} &= -(\mathbf{y} - \tilde{\boldsymbol{\mu}})^T \left(\tilde{\mathbf{\Psi}} + \mathbf{\Sigma}^2\right)^{-1} \frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{\theta}_i^*} \\ &- \frac{1}{2}\text{tr}\left(\left(\tilde{\mathbf{\Psi}} + \mathbf{\Sigma}^2\right)^{-1} \frac{\partial \tilde{\mathbf{\Psi}}}{\partial \boldsymbol{\theta}_i^*} + (\mathbf{y} - \tilde{\boldsymbol{\mu}})(\mathbf{y} - \tilde{\boldsymbol{\mu}})^T \left(\tilde{\mathbf{\Psi}} + \mathbf{\Sigma}^2\right)^{-1} \frac{\partial \tilde{\mathbf{\Psi}}}{\partial \boldsymbol{\theta}_i^*} \left(\tilde{\mathbf{\Psi}} + \mathbf{\Sigma}^2\right)^{-1}\right) \end{aligned} \tag{25}$$

The derivatives of the LMC mean and variance functions (22) are:

$$\begin{cases} \dfrac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{\theta}_i^*} = \tilde{\mathbf{B}} \dfrac{\partial \mathbf{A}}{\partial \boldsymbol{\theta}_i^*} \mathbf{m} \\ \dfrac{\partial \tilde{\mathbf{\Psi}}}{\partial \boldsymbol{\theta}_i^*} = \tilde{\mathbf{B}} \times \text{diag}\left(\left(\dfrac{\partial \mathbf{A}}{\partial \boldsymbol{\theta}_i^*}\right)^T \left(\mathbf{S} - \mathbf{K}^{\mathbf{z},\mathbf{z}}\right)\mathbf{A} + \mathbf{A}^T\left(\mathbf{S} - \mathbf{K}^{\mathbf{z},\mathbf{z}}\right)\left(\dfrac{\partial \mathbf{A}}{\partial \boldsymbol{\theta}_i^*}\right)\right)\tilde{\mathbf{B}}^T \end{cases} \tag{26}$$

Given the definitions of $\mathbf{A}$ in (15), we finally have:

$$\frac{\partial \mathbf{A}}{\partial \boldsymbol{\theta}_i^*} = \text{bdiag}\left(\left\{\frac{\partial \mathbf{A}_v}{\partial \boldsymbol{\theta}_i^*}\right\}\right) = \text{bdiag}\left(\left\{\frac{\partial \mathbf{K}_v^{\mathbf{x}^*,\mathbf{z}}}{\partial \boldsymbol{\theta}_i^*}\left(\mathbf{K}_v^{\mathbf{z},\mathbf{z}}\right)^{-1}\right\}\right) \tag{27}$$

In practice it is more efficient to use automatic differentiation to obtain the gradient, rather than attempting to implement these derivations directly, however they establish that the gradient (25) exists and is consistent.

3. Bayesian estimation with GP regression surrogate

3.1. Training sample design and GP surrogate training

The generation of the training data $\mathbf{X}$ from the simulation model needs to be planned carefully. We impose a computationally constrained setting that requires this training data be simulated prior to training the LMC surrogate, and as established in section 2.2, the latter's convergence depends on the properties of that pre-existing training data. This impacts on four decisions: setting bounds $\left[\underline{\theta}, \overline{\theta}\right]$ on the simulation model's underlying parameter space Θ, choosing a design for drawing training samples $\boldsymbol{\theta}_s$ from Θ, setting the number of samples S to draw and the length T of the time-series simulation for each $\boldsymbol{\theta}_s$.

The need for a set of bounds on Θ stems from the fact that the universal approximation property holds on a compact subset $\mathcal{Z}$ of the full input space $\mathcal{X}$, assumed to be $\mathbb{R}^d$ in our case. Because $\mathcal{X}$ contains Θ, picking $\mathcal{Z} \subseteq \mathcal{X}$ involves picking a bounded interval for Θ. Choosing the bounds will require the researcher's domain knowledge of the simulation model, such as the nature of the parameter involved (such as a share, a ratio, etc.), or prior estimates from the literature. Similarly, a critical factor in the choice of design for

drawing the training samples is knowledge of the variability of the mean one-step ahead behaviour of the model over $\mathcal{Z}$. In practical terms, the convergence of the predictions is mainly controlled by ℓ_{v^*}, the smallest length scale from the V latent RBF kernels: if the one-step ahead model predictions are very sensitive to certain parameter values, a finer sample of points will be required to obtain a reliable surrogate, affecting either the number of samples required, or the chosen design. For instance, if model predictions are discontinuous in a given region of Θ, it might be beneficial to pick an adaptive design that samples those regions more densely. This is where Bayesian optimization, which enables online acquisition of the training samples as the GP is trained, can offer an advantage.

While a large literature already details the available design options and their relative benefits (see for instance Santner et al., 2018), a few aspects need to be detailed. First, the computational constraints on model simulations imply that the chosen design must have good space-filling properties (Lamperti et al., 2018). Santner et al. (2018) recommends the Latin hypercube design (LHD) for this context, while Salle and Yıldızoğlu (2014) provide evidence that the additional orthogonality of the near-orthogonal Latin hypercube (NOLH) design (Cioppa and Lucas, 2007) improves prediction accuracy of GP surrogates. The drawbacks of the LHD and NOLH are their fixed size, limiting the ability to increase the size of the training data if required. In addition, even if samples $\boldsymbol{\theta}$ are orthogonal, because the training data (4) contains both $\boldsymbol{\theta}$ and lagged time-series data $\mathbf{Y}$, $\mathbf{X}$ will be not orthogonal in general, limiting the attractiveness of NOLH. Sobol sequences can instead be extended easily, and while Liefvendahl and Stocki (2006); Santner et al. (2018) show that they lead to less precise predictions compared to LHD, due to a greater range of inter-point distances, the practical difference between the two is minimal, especially as S increases. Finally, a Sobol design allows an initial simulation run that has insufficient S for LMC convergence to be augmented ex-post. This potentially allows for a more active sampling strategy, where additional training samples can be drawn from targeted locations in Θ, using for example Bayesian optimization. This is not implemented in this work, as the aim is to establish whether BEGRS can still perform in the most constrained case, but is discussed in the conclusion.

In addition to picking a design, one needs to choose the number of samples S and the length of the simulated time-series T, which together determine the number of training observations $N = S(T-1)$. First, N should be as large as possible given the computational constraint, in order to improve the convergence of the surrogate. Second, S should also be as large as possible, to ensure good space-filling of Θ. The strategy adopted for the applications in section 4 is to pick T to match the number of observations in the empirical data, and infer S from the feasible number of T-length simulations given the computational budget.

Once the training data $\mathbf{X}$ is available, the next step is to pick the number of latent variables V to use in the LMC and the number of inducing points $\mathbf{Z}$ in the variational approximation. V can be set using principal component analysis (PCA) or a factor analysis of the simulated data $\mathbf{Y}$ to determine the minimal number of latent variables required to summarize the data. Equation (6) implies that $\mathbf{Y}$ contains additive variable-specific noise on top of the LMC predictions $\tilde{\mathbf{f}}$, suggesting that factor analysis might be preferable to the strict orthogonal decomposition of $\mathbf{Y}$ provided by PCA. Note that in either case, the loadings obtained will generally not match the LMC loadings $\mathbf{B}$. This is because the latent variables in the LMC capture the correlations in the training inputs $\mathbf{X}$ at different length scales ℓ_v, a constraint that factor analysis or PCA do not possess.

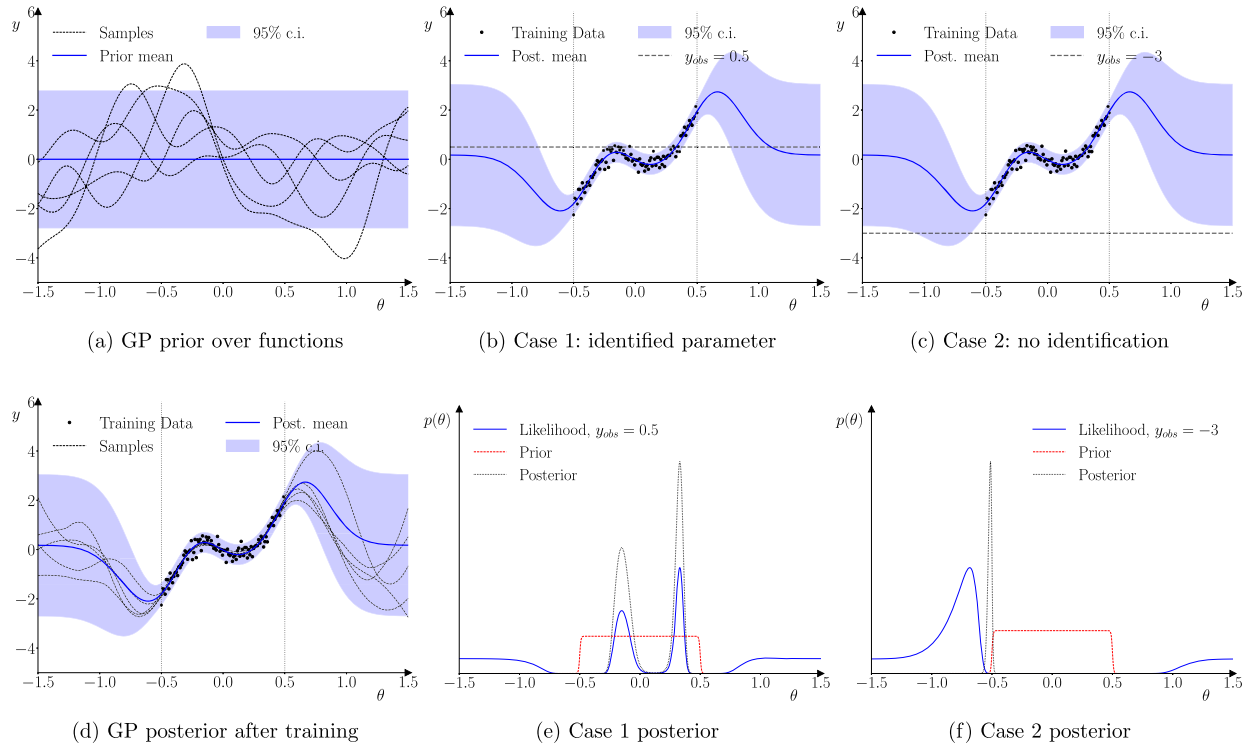
Less guidance is available for picking the number of inducing points. The $\mathcal{O}\left((\log N)^d\right)$ bound of Burt et al. (2019, 2020) is a scaling guarantee, not a method of calculating the number of inducing points required. In addition, if the d-dimensional $\mathbf{X}$ falls on a lower-dimensional manifold embedded in $\mathcal{X}$, then the Burt et al. (2019, 2020) bound is driven by the dimensions of that manifold. This is likely here, given that $\mathbf{X}$ contains lagged values of $\mathbf{Y}$, itself assumed to be reducible to a smaller number of latent variables V. The variational approximation will converge once a sufficient number of inducing points has been included, therefore the practical process suggested in the literature is to run the GP regression several times with an increasing number of inducing points, and identify the threshold at which adding points no longer improves the ELBO (20).

Given that the LMC surrogate will only be useful if it converges to the predictions of the first-order Markov approximation (3), and given the various issues listed here that may affect this convergence, it is recommended that the convergence of the BEGRS posterior be assessed using the simulated Bayesian computation (SBC) approach of Talts et al. (2018), which itself extends the simulated validation method proposed by Cook et al. (2006). This enables to identify any critical convergence issue in the first (training) stage of BEGRS, prior to carrying out the estimation on empirical data.

3.2. Minimal prior, posterior estimation and identification

The purpose of the GP surrogate is to provide a low-cost approximation of the simulation model's likelihood $p(\mathbf{y}^*|\boldsymbol{\theta}^*)$, given empirical data $\mathbf{y}^*$ and a candidate parameter vector $\boldsymbol{\theta}^*$, enabling the use of Bayesian methods. The availability of the surrogate likelihood's gradient (25) means that assuming a differentiable prior $\boldsymbol{\theta}^*$, the posterior can be explored with gradient-based methods, such as Hamiltonian Monte-Carlo (HMC).

While the specific choice of parameter prior $p(\boldsymbol{\theta}^*)$ is up to the researcher, the use of a GP regression surrogate imposes a minimal requirement. GP regression converges to the true f_0 in a compact subset $\mathcal{Z}$ of the input space $\mathcal{X}$, set using bounds $\left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]$ on the parameter space Θ. The kernel (7), however, is defined over $\mathcal{X}$ and serves as the GP's prior over the space of all continuous functions $C(\mathcal{X})$. This is illustrated in Fig. 1(a), where the GP prior is defined over $\mathbb{R}$, whereas in Fig. 1(d) the GP posterior only converges to the true data-generating process in the bounded interval containing the training data. Outside of that interval, the GP prior entirely determines the GP posterior. This results in a surrogate likelihood defined over all of $\mathbb{R}^{d_\Theta}$ but trained only for values of $\boldsymbol{\theta} \in \left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]$. In principle, a continuous uniform prior over $\left[\underline{\boldsymbol{\theta}}, \overline{\boldsymbol{\theta}}\right]$ is sufficient to restrict estimates for $\boldsymbol{\theta}^*$ within those bounds. In practice, however, the discontinuity at the boundary creates problems for gradient-based algorithms. Instead, to ensure that the gradient of the prior is defined at the boundary of the parameter space itself, we recommend using a smooth relaxation of the uniform distribution in the



(a) GP prior over functions (b) Case 1: identified parameter (c) Case 2: no identification

(d) GP posterior after training (e) Case 1 posterior (f) Case 2 posterior

Fig. 1. Illustration of surrogate GP likelihood. (For interpretation of the colours in the figures, the reader is referred to the web version of this article.)

form of double sigmoid function, where $\alpha > 0$ controls the slope at the boundary. This ensures that estimates of $\boldsymbol{\theta}^*$ remain inside the bounds, while also ensuring that gradient-based algorithms do not stall on the bounds themselves. The following log prior and log prior gradient can be used alongside the log-likelihood and its gradient.

$$\begin{cases} \ln p(\boldsymbol{\theta}^*) = -\sum_i \left(\ln\left(1 + e^{-\alpha(\theta_i^* - \underline{\theta})}\right) + \ln\left(1 + e^{-\alpha\left(\overline{\theta} - \theta_i^*\right)}\right)\right) \\ \dfrac{\partial \ln p(\boldsymbol{\theta}^*)}{\partial \theta_i^*} = \alpha\left(\dfrac{e^{-\alpha(\theta_i^* - \underline{\theta})}}{1 + e^{-\alpha(\theta_i^* - \underline{\theta})}} + \dfrac{e^{-\alpha\left(\overline{\theta} - \theta_i^*\right)}}{1 + e^{-\alpha\left(\overline{\theta} - \theta_i^*\right)}} \right) \end{cases} \tag{28}$$

Even with a well-defined prior gradient on the parameter space boundaries, the fact that the LMC surrogate is defined on a subset $\mathcal{Z}$ of the full input space $\mathcal{X}$ creates an identification issue specific to BEGRS, in addition to existing identification issues, such as flat or multi-modal likelihoods, or lack of convergence during LMC training. Figs. 1(b) and (e) illustrate the well-behaved case where an empirical observation is consistent with the range of simulated outputs in the training data. The resulting surrogate likelihood remains non-vanishing outside of the parameter boundaries, but the posterior has two clear modes consistent with the likelihood. In Figs. 1(c) and (f) the empirical observation is instead out of line with the training data. Here the diffuse GP prior provides a better fit than the GP posterior, pulling the mode of the likelihood outside of the bounds $\left[\underline{\theta}, \overline{\theta}\right]$. The application of the minimal prior (28) results in a distinctive degenerate distribution at the parameter boundary itself, shown in Fig. 1(f).

More informative priors can also be used, the only requirement being that they are continuous across the bounds $\left[\underline{\theta}, \overline{\theta}\right]$ and vanishing outside of them. One must take care, however, when integrating prior information, that the resulting prior does not overpower the surrogate likelihood. This will potentially be flatter than the true, unobserved, likelihood of the model, due to the fact that the GP prediction contains a prediction error in addition to the standard noise term. This prediction error might be sizeable in the case where the computational constraint restricts the amount of training data available.

4. Applications

Two applications are provided to illustrate the Bayesian estimation with Gaussian regression surrogate (BEGRS) framework, both of which were carried out using the companion Python toolbox developed for the methodology. The toolbox uses the GPytorch implementation of GPs of Gardner et al. (2018) and is available from github.com/Sylvain-Barde/begrs. The code for replicating the VARMA and ABM exercises is available from github.com/Sylvain-Barde/begrs_varma and github.com/Sylvain-Barde/begrs_sfc respectively.

4.1. Parameter recovery on known data generating process

This first application aims to verify that the BEGRS framework can indeed produce consistent posteriors with a small number of simulation runs relative to the dimensionality of the parameter space. This is done by running a Monte Carlo exercise where BEGRS is used to estimate the $\boldsymbol{A}$ and $\boldsymbol{B}$ matrices from the following VARMA(1,1) specification:

$$X_t = \boldsymbol{A}X_{t-1} + \eta_t + \boldsymbol{B}\eta_{t-1}, \qquad \eta_t \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \Sigma), \qquad \begin{cases} \Sigma_{i,i} = 1 \\ \Sigma_{i,j} = \rho < 1, \quad i \neq j \end{cases} \tag{29}$$

This specification is picked because VARMA models are known to be challenging to estimate in the first place (see for instance Lütkepohl, 2005; Dias and Kapetanios, 2018), and the Vecchia approximation (3) used in the BEGRS likelihood function will discard information compared with the full likelihood for (29), as this specification is not a first-order Markov process. In addition to the full VARMA(1,1) estimation, in a second setting $\boldsymbol{B}$ is set to a null matrix, reducing (29) to a VAR(1), which does possess the first-order Markov property. This results in two experimental settings, one where BEGRS is expected to do well by construction, and a second that is more challenging, as explicitly misspecified.

In order to keep the number of estimated parameters similar in both cases, we set the number of observable variables to $M = 3$ in the VARMA(1,1) case and $M = 4$ in the VAR(1) case, leading respectively to 18 and 16 parameters to be estimated. Only $\boldsymbol{A}$ and $\boldsymbol{B}$ are estimated via BEGRS, as the additive noise η_t is captured directly during the training of the GP regression. Instead, the level of correlation between variables ρ forms part of the setting of the Monte Carlo exercise, by verifying that the LMC can cope with correlated additive noise. In each case, the training dataset consists of $S = 1000$ series of $T = 200$ observations, resulting in $N = S(T-1) = 199{,}000$ training observations. The training series were each generated with distinct $\boldsymbol{A}^s$ and $\boldsymbol{B}^s$ matrices, with individual $A^s_{i,j}$ and $B^s_{i,j}$ values drawn from the $[-0.7, 0.7]$ range using of Sobol sequences. Each $\boldsymbol{A}^s$ and $\boldsymbol{B}^s$ was checked to ensure that their eigenvalues lie within the unit circle. 1233 and 1166 Sobol draws were required to obtain 1000 stable parametrizations in the VAR(1) and VARMA(1,1) cases respectively.

A testing set was generated by drawing additional $S = 1000$ stable Sobol parameterizations and simulating them for $T = 200$ observations. The use of the same Sobol sequence as the training data ensures that the testing parameters samples set are located in between the training samples and therefore distinct from them. In a first analysis, one of these parameterizations was used as empirical data in a parameter recovery exercise, using both BEGRS and a standard benchmark, approximate Bayesian computation with sequential Monte Carlo (ABC-SMC). The ABC-SMC estimations all used 5000 particles iterated over 40 generations. The fastest estimation required 966,596 simulation calls, the slowest 1,561,513, to be compared with the 1,000 simulations used by BEGRS. As a second step a SBC diagnostic was run on the full testing set, in order to provide a more general evaluation. Both analyses used 60, 125 and 250 inducing points as a sensitivity check for the surrogate training.

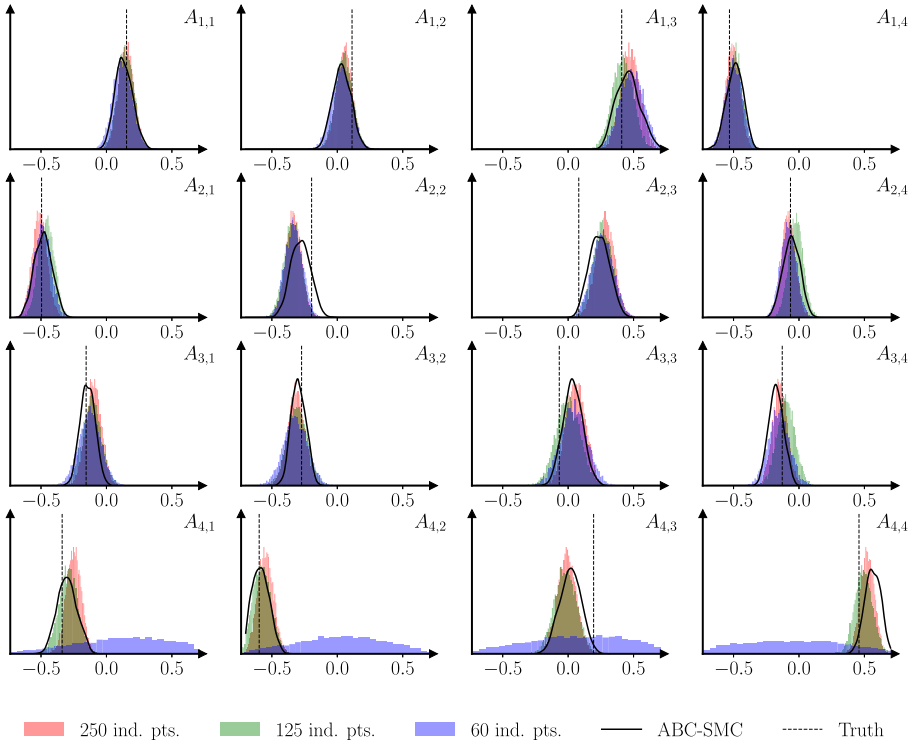
The main results for both analyses are presented in Figs. 2 and 3 for the $\rho = 0$ case. Results for the $\rho = 0.25$ and $\rho = 0.5$ cases, which are very similar, are provided in online appendix B. In the case of the VAR(1), where the Vecchia approximation (3) to the likelihood is valid, Fig. 2(a) shows that BEGRS can perform on par with ABC-SMC, using a simulation budget that is three orders of magnitude smaller. Performance degrades in the 60 inducing point setting, but in a graceful way: rather than affecting all estimates, consistency fails for the last row of $\boldsymbol{A}$ but is maintained for others. This is symptomatic of the GP surrogate being unable to provide good predictions for the corresponding variable, suggesting that the LMC has not converged for that variable. The SBC analysis in Fig. 2(b) shows that this conclusion holds more generally for the full posterior.

As expected, the VARMA(1,1) setting is more challenging. Several key findings emerge from the posterior distributions in Fig. 3(a) and (b). First, the intrinsic difficulty in estimating the parameters is visible from the much wider distributions produced by ABC-SMC. Second, the BEGRS posterior distributions often recover the parameters and sometimes match those obtained with ABC-SMC. However, they also fail to recover the true parameter values in several cases, seemingly because they are narrower compared to the ABC-SMC benchmark. Again, the SBC analysis confirms this, as the very pronounced U-shape of the rank histograms, which remains even after thinning the posterior samples to eliminate autocorrelation, is characteristic of a posterior that is too narrow.

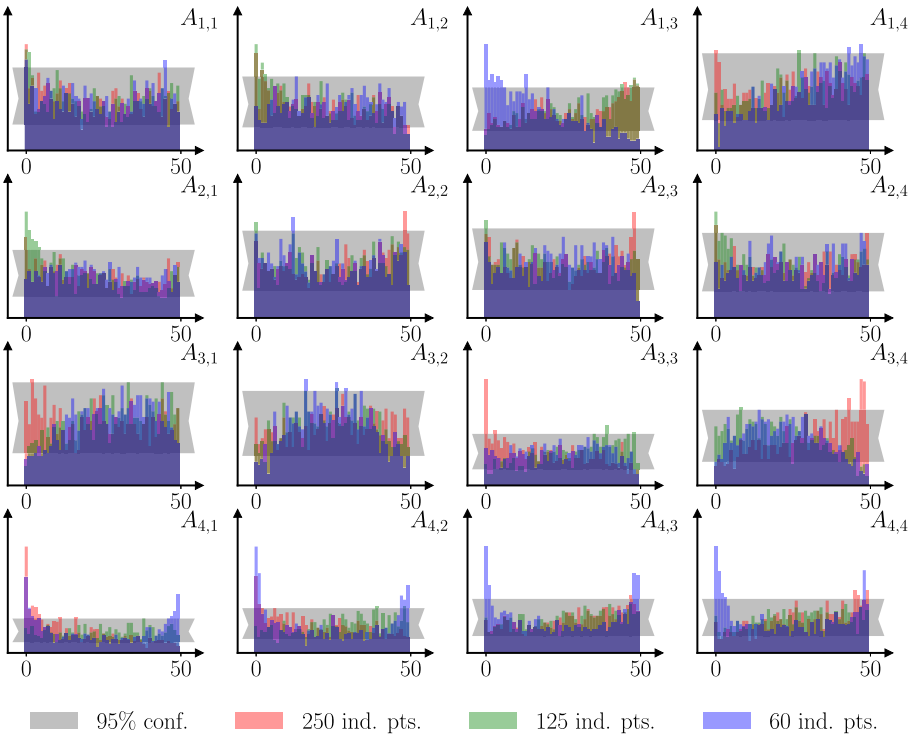
4.2. Empirical application to a large-scale ABM

The second application is an empirical estimation of the free parameters of the Caiani et al. (2016) model, offering a good illustration of the type of computationally constrained setting that BEGRS is designed for. This model was developed following the financial crisis of 2008-2009 to improve understanding of the endogenous emergence of financial shocks and their contagion to the real economy. The model combines a stock-flow consistent approach with fully-fledged commercial and financial networks between households, banks and vertically differentiated firms. While this creates high computational requirements, it enables both the replication on aggregate of the deep recessions that follow severe financial crises, as well as the analysis at a disaggregated level of the mechanisms that drive them. The model has been used to analyse the effect of fiscal policy design on inequality (Caiani et al., 2019), the effect of fiscal targets in a monetary union (Caiani et al., 2018) and the transmission of monetary policy (Schasfoort et al., 2017).

The model's computational requirements also make it a valuable testbed for validation methodologies aimed at large-scale models, for example simulated model selection (Barde, 2020). Full estimation of the model's free parameters from empirical data has never been carried out, instead the literature has relied on replication of stylized facts combined with coarse-grid sensitivity analyses on a small subset of the free parameters. By contrast, we show here that BEGRS can successfully estimate all the free parameters from US

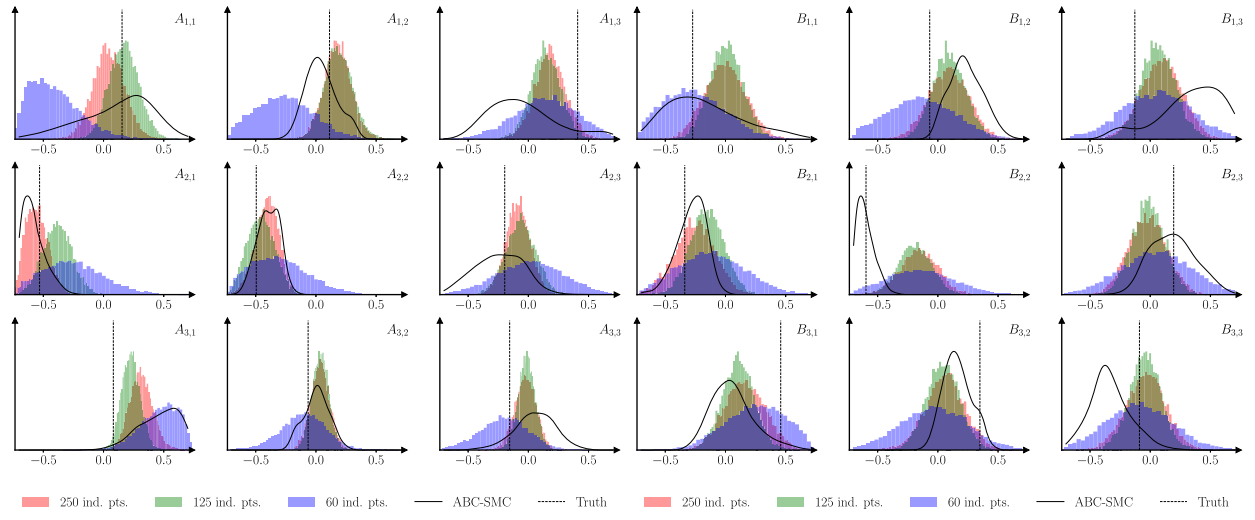


(a) BEGRS-NUTS estimates



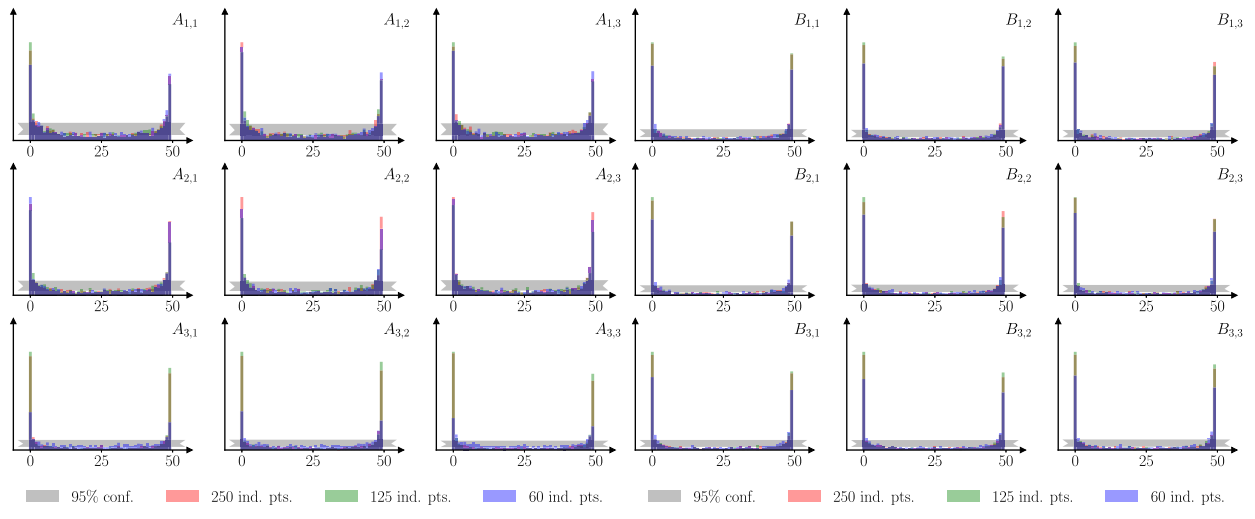
(b) SBC rank histogram

Fig. 2. Results for VAR matrix A, $\rho = 0$.



(a) BEGRS-NUTS estimates, matrix A

(b) BEGRS-NUTS estimates, matrix B



(c) SBC rank histogram, matrix A

(d) SBC rank histogram, matrix B

Fig. 3. Results for VARMA setting, $\rho = 0$.

macroeconomic time-series data, using only 1000 training simulation runs. In order to illustrate the ability of BEGRS to re-use the same surrogate in multiple empirical estimations, the analysis uses two distinct datasets. The first is the Smets and Wouters (2007) data set, with $T = 160$ quarterly observations for the deviation of labour hours from trend L, the real policy rate r, the rate of inflation π, and the real first log difference of output Δy, consumption Δc, investment Δi and wages Δw from 1965 to 2004. The second is a shorter ($T = 82$) dataset covering the 1997-2017 period and containing the 1997, 2001 and 2008 crises, and the subsequent period of low inflation and near-zero interest rates.

The model has 12 free parameters, presented in Table 1, that are not set from direct observation or inferred from the steady-state constraints. The bounds on the first 5 parameters are those of the Caiani et al. (2016) sensitivity analysis, while the bounds on the remaining 7 parameters are set to allow reasonable variation around the value used in the original paper. In cases where the parameter is naturally restricted to $[0, 1]$ (such as λ and ι), the bounds were offset from those natural bounds to avoid pathological behaviour in the simulations. Two simulated datasets were generated, one for training the GP surrogate and the other for a SBC diagnostic. These both contain $S = 1000$ series of $T = 200$ observations, each using samples drawn from the same 12-dimensional Sobol sequence. Each simulation run contains 500 observations, including a 300 observation burn-in. The average wall time per run was 44 minutes, and the total wall time for $S = 1000$ using a 36-core HPC node was 21 hours. The LMC surrogate was trained using $V = 4$ latent variables and 250 inducing points, and estimations used the minimal prior (28), with $\alpha = 20$.

The last four columns of Table 1 show two sets of BEGRS estimates for each of the empirical datasets: the mode of the posterior, obtained using BFGS, and the mean of the posterior, estimated from 10,000 NUTS iterations. While the estimates for some parameters,

Table 1
Caiani et al. (2016) free parameter estimates.

Parameter		Base value	Prior range	Posterior estimates			
				SW data		Crisis data	
				Mode	Mean	Mode	Mean
Bank risk aversion (C firms)	ζ_c	3.922	1 - 10	4.795	4.758	4.421	4.719
Bank risk aversion (K firms)	ζ_k	21.513	5 - 40	14.992	16.213	18.063	21.291
Profit weight in firm inv.	γ_1	0.010	0.01 - 0.04	0.012	0.014	0.019	0.022
Capacity weight in firm inv.	γ_2	0.020	0.01 - 0.04	0.032	0.031	0.035	0.031
C firm precaution deposits	σ	1.000	0.5 - 1.5	1.213	1.154	1.070	1.031
Intens. of choice - C/K markets	ε^{CK}	0.150	0.05 - 0.3	0.116	0.126	0.159	0.167
Intens. of choice - credit/deposit	ε^{cd}	0.200	0.05 - 0.3	0.271	0.259	0.278	0.247
Adaptive expectation param.	λ	0.250	0.1 - 0.8	0.697	0.716	0.754	0.713
Labour turnover ratio	ϑ	0.050	0.025 - 0.15	0.090	0.092	0.103	0.098
Folded normal std. dev.	σ^2_{FN}	0.009	0.005 - 0.015	0.014	0.014	0.012	0.012
Haircut param. - firm default	ι	0.500	0.3 - 0.7	0.435	0.441	0.557	0.520
Unemp. threshold - wage rev.	ν	0.080	0.05 - 0.11	0.071	0.074	0.073	0.076

Table 2
Caiani et al. (2016) MIC goodness-of-fit analysis.

	L	r	π	Δy	Δc	Δi	Δw	Aggr.	-ln P
Smets & Wouters dataset (1965:Q1 - 2004:Q4)									
Original	949.98	2617.81	1903.02	937.05	1027.53	924.44	1554.42	10096.43	2209.27
Mode	943.68	1617.06	1050.68	902.15	905.48	1013.42	1151.00	8110.00	2120.32
Mean	948.84	1626.94	1026.57	932.55	906.05	1040.81	1167.65	8135.47	2136.24
Crisis period (1997:Q1 - 2017:Q2)									
Original	574.71	775.88	358.82	407.90	472.76	463.66	999.78	4313.37	1207.75
Mode	653.86	564.57	438.61	464.21	405.88	506.44	597.00	3756.72	1175.67
Mean	645.45	584.32	448.90	471.48	420.61	513.76	576.34	3735.01	1195.50

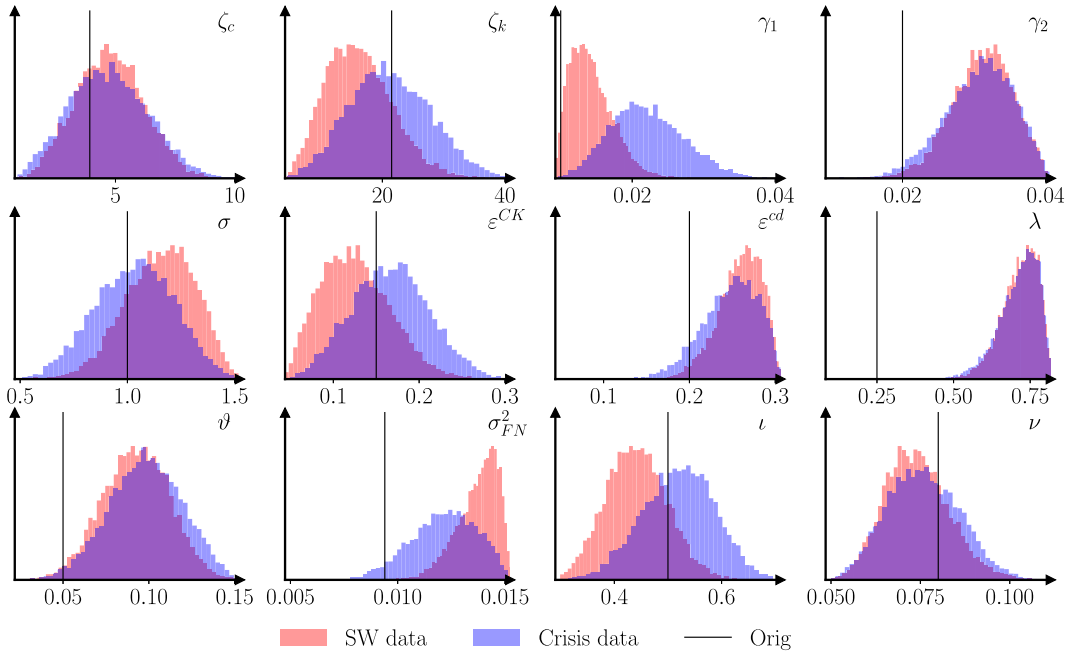
Note: $-\ln P$ is the negative log posterior provided by the surrogate for a parameterization. The aggregate MIC score over the dataset is not the sum of the variable-level scores, as the former discards the mutual information between variables to avoid double-counting, and its scale is not directly comparable to $-\ln P$.

such as the risk aversion ζ_c, the intensities of choice ε^{CK}, haircut parameter ι or the unemployment threshold ν are close to the values used in by the original authors, others diverge clearly, like the expectation parameter λ or folded normal parameter σ^2_{FN}. The marginal frequencies of the NUTS iterations are shown in Figs. 4(a) and confirm that estimates for λ, and to a lesser extent and γ_1 and σ^2_{FN} lie very close to the boundaries of the parameter. The σ^2_{FN} parameter in particular probably suffers from the specific identification problem discussed in section 3.2. Fig. 4(b) presents the corresponding SBC analysis. The rank distributions for all parameters clearly depart from uniformity, with the characteristic U-shape that indicates under-dispersion of the posterior. This deviation is not as severe, however, as the one exhibited by the VARMA estimation in Fig. 4(c) and (d), which suggests that the surrogate, while imperfect, might nevertheless be useful.

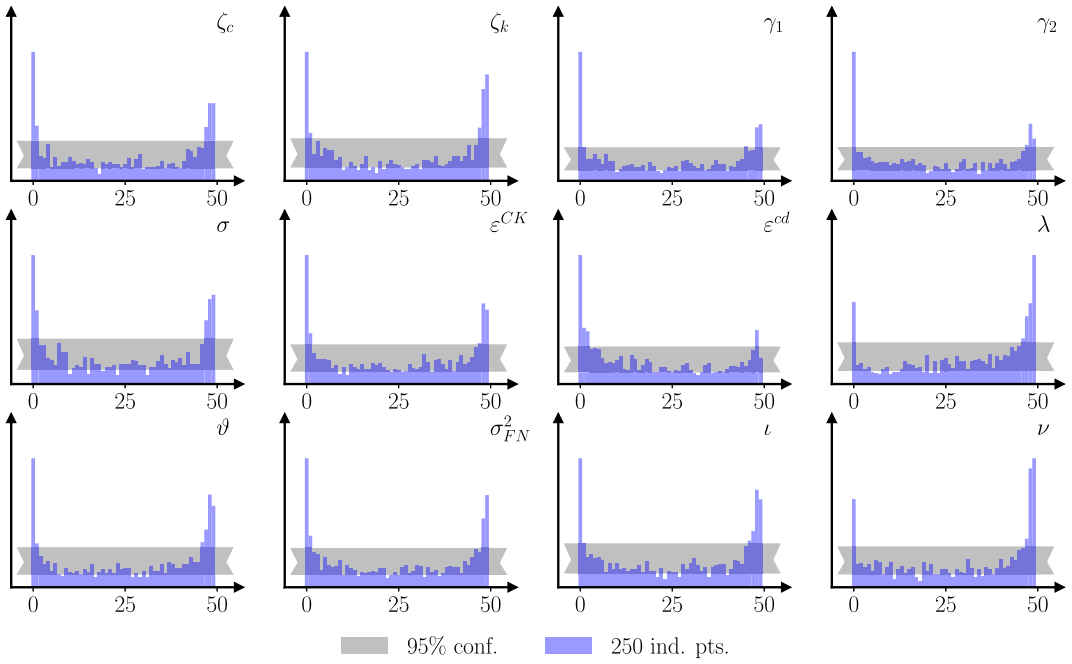
Given these potential identification and convergence problems, as well as the relatively wide marginal distributions on some parameters, one may question whether the BEGRS estimates improve on the original calibration. In order to evaluate this we first replicate the analysis of Barde (2020) and use the multivariate Markov information criterion (MIC) of Barde (2017) for both the original calibration and the BEGRS estimates. The MIC is an extension of the Akaike information criterion (AIC) to simulation models, in that it provides an unbiased estimate of the cross-entropy between a set of simulated and empirical data series. Relative MIC scores across simulations thus provide a measurement of the relative KL divergence between the simulation models on a given empirical dataset. The scores obtained for each calibration, both at the variable and aggregate levels, are provided in Table 2, alongside the (negative) log-posterior provided by the BEGRS surrogate in the last column. The aggregate MIC score is in agreement with the log posterior, and confirms that the BEGRS estimates significantly improve the aggregate goodness-of-fit of the model on both the Smets and Wouters (2007) and crisis datasets compared to the original calibration.

At the variable level, Barde (2020) showed that for the Smets and Wouters (2007) dataset, the original calibration was characterized by reasonable performance on the aggregate quantities (L, Δy, Δc and Δi) but a very poor fit on the aggregate prices (r, π and Δw). Both sets of BEGRS estimates improve the fit on these variables, at the cost of a slightly worse performance on Δi. The situation is slightly different on the crisis dataset, where the fit on r and Δw still improves, but worsens on nearly all the other variables, particularly inflation π.

In order to illustrate the origins of these trade-offs in fit, Fig. 5 plots the unconditional densities of the observable variables produced by the simulation model for the various parameterizations, below box plots of the corresponding empirical data. These reveal that the poor fit of the original ABM calibration on the Smets and Wouters (2007) data stems from the fact it produces distributions that are narrower than their empirical counterparts, particularly on r, π and Δw. By contrast, the BEGRS estimates produce wider distributions on all variables, improving the fit on the SW dataset with the exception of investment Δi. In this case, the



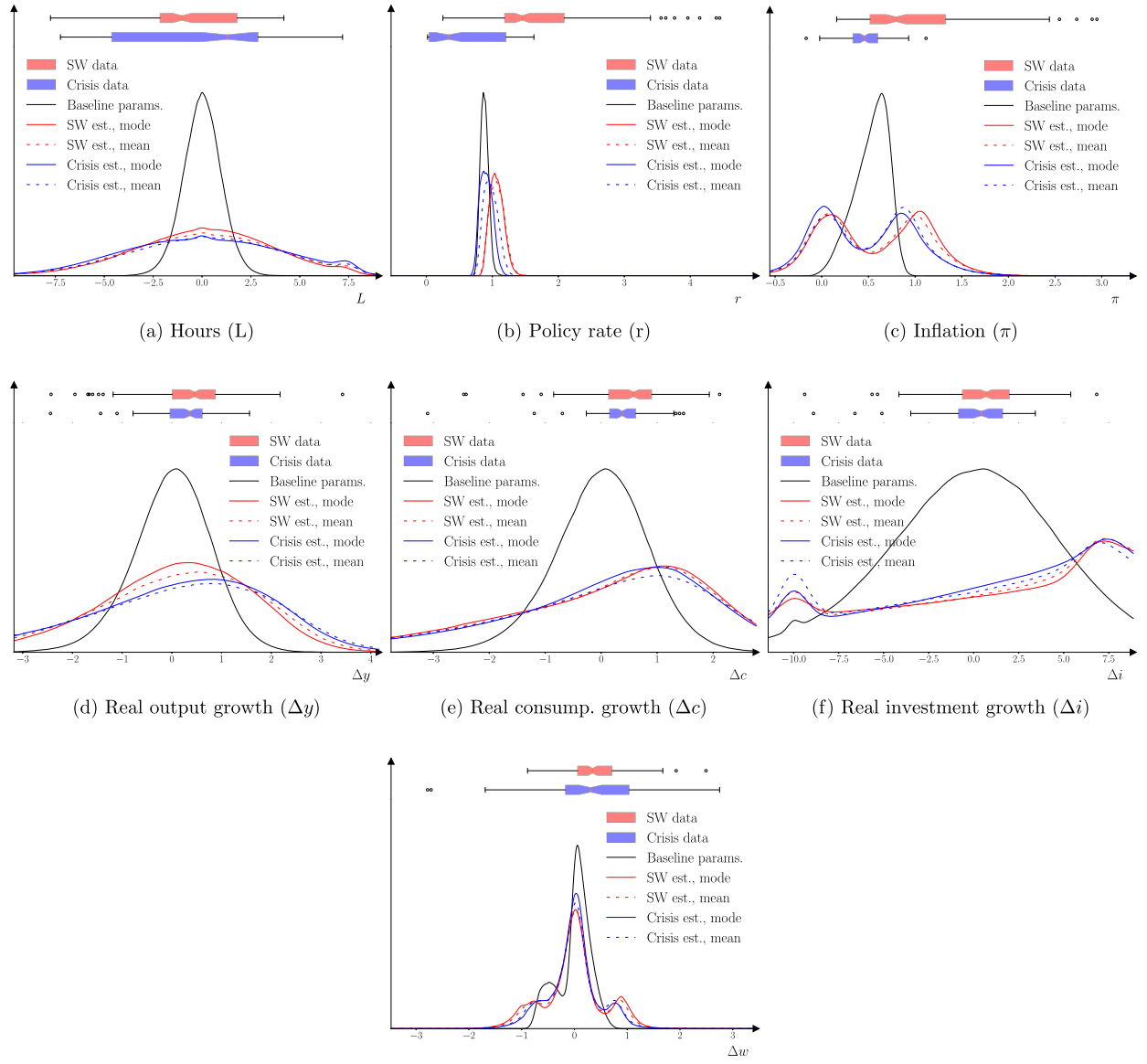
(a) BEGRS-NUTS estimates



(b) SBC rank histogram

Fig. 4. BEGRS results for US macroeconomic estimation.

original calibration already provides a relatively wide distribution, and the wider distribution brought on by the BEGRS estimates is detrimental to the fit. Similarly, the poorer fit of certain variables on the crisis dataset is itself explained by the fact that their empirical distributions are much narrower, to the point that for certain variables (such as π or Δi), the original calibration outperforms the wider BEGRS calibrations. Finally, the multimodality of some distributions reveal that these wider simulated distributions are achieved by the model switching between two regimes, one with higher inflation, growing investment and real wages, and one with falling investment and wages and zero or negative inflation.



(a) Hours (L) (b) Policy rate (r) (c) Inflation (π)

(d) Real output growth (Δy) (e) Real consump. growth (Δc) (f) Real investment growth (Δi)

(g) Real wage growth (Δw)

Fig. 5. Unconditional densities of simulated data for observable variables.

5. Conclusion

This paper presents and tests a Bayesian estimation framework specifically designed for computationally demanding time-series simulation models, in order to make the best use of a potentially limited amount of training data. Core to this are the use of a one-step-ahead predictor, in the form of a first-order Markov approximation, which leverages the time-dimension of the simulated data to increase the number of training observations, and the use of GP regression as the surrogate model. This choice is driven by the desirable theoretical properties of the GP surrogate, notably the universal approximation property, proven converge with minimal assumptions on the underlying model, and the self-regularization property of the GP estimation, which limits overfitting on potentially limited training data. The methodology's two-stage nature ensures that the same surrogate can be re-used for different empirical datasets, further amortizing costly simulations. The paper extends a pre-existing result on GP regression to show that the method is consistent, and illustrates its functionality by providing the first existing empirical estimation of all the free parameters of a large-scale, computationally intensive ABM.

Several limits merit highlighting. First, the BEGRS framework nests two distinct Bayesian estimations, the first estimating the GP parameters from the training data, the second using the resulting GP surrogate to estimate the simulation model parameters from the empirical data. In addition to the standard identification concerns on the model parameters, one also needs to verify that the

surrogate likelihood generated by the GP is itself valid. This involves checking that the first-stage estimation of the GP has converged, but also verifying the validity of the first-order Markov approximation. A key recommendation is to always run a SBC analysis as a diagnosis tool on the posterior, to assess whether the BEGRS surrogate accurately approximates the model's behaviour. Second, the BEGRS estimation framework does not offer a panacea: a large amount of simulation data is still required, increasingly so as the dimensionality of the parameter space d_Θ grows beyond the baseline of 10-20 parameters used here.

These caveats suggest two directions for future research. In principle, the convergence result obtained here extends to higher-order Markov processes, which means that the methodology could be extended to training data containing longer memories, at the cost of a larger input space for the GP. A first objective, therefore, is to investigate dimensionality reduction methods that can be applied to larger input spaces, caused either by larger parameter spaces or more lags of memory. A second direction is the inclusion of active acquisition strategies for drawing training samples from the parameter space, along the lines of Bayesian optimisation. In order to maintain the re-usability property on multiple empirical datasets, the acquisition function cannot depend on empirical data, for example focusing on regions with high likelihoods. Instead, an interesting option is that used in Lamperti et al. (2018), where additional training parameter samples are drawn from regions of the parameter space where the predictive performance of the surrogate on the training data itself is poor. This can potentially further reduce the simulation requirements, particularly when the simulation model's output response is discontinuous across the input space, slowing the convergence of the current approach.

Acknowledgements

The author is grateful to participants at the WEHIA 2021, CEF 2021 and CEF 2022 conferences for their comments. Particular thanks goes to Herbert Dawid, Christophre Georges, Blake LeBaron and Junior Maih for their input on a preliminary version of the manuscript, as well as the associate editor and three anonymous referees who greatly improved the analysis with their suggestions. The work carried out also benefited from the use of the specialist and High Performance Computing systems provided by Information Services at the University of Kent, specifically the ICARUS HPC cluster. Any errors in the manuscript remain of course the author's.

Appendix A. Summary of notation

Table 3
List of variables.

Vectors		Matrices	
Symbol	Definition	Symbol	Definition
$\mathbf{y}$	Vectorised observable variables	$\mathbf{Y}$	Observable variables
$\boldsymbol{\theta}$	Model parameters	$\mathbf{X}$	GP inputs
$\boldsymbol{\sigma}$	Additive noise per variable	$\boldsymbol{\Sigma}^2$	GP additive noise
$\boldsymbol{\mu}$	GP mean prediction	$\boldsymbol{\Psi}$	GP covariance
$\boldsymbol{\ell}$	Kernel lengthscales	$\mathbf{K}^{\mathbf{x},\mathbf{x}}$	Kernel covariance
$\mathbf{f}$	GP prediction	$\mathbf{B}$	LMC loadings
$\mathbf{u}$	Inducing values	$\mathbf{Z}$	Inducing points
$\mathbf{m}$	Inducing value means	$\mathbf{S}$	Inducing value covariance

Table 4
Summary of operations on vectors $\mathbf{x}$, matrices $\mathbf{X}$.

Symbol	Definition	Vector operation	Matrix operation
$\mathbf{x}^*$, $\mathbf{X}^*$	empirical variable	-	-
$\mathbf{x}_v$, $\mathbf{X}_v$	Individual latent-level variable	-	-
$\{\mathbf{x}_v\}$, $\{\mathbf{X}_v\}$	Set of V latent-level variables	$\{\mathbf{x}_1, \mathbf{x}_1, \dots, \mathbf{x}_V\}$	$\{\mathbf{X}_1, \mathbf{X}_1, \dots, \mathbf{X}_V\}$
$\mathbf{x}$, $\mathbf{X}$	vectorised variable	$\mathbf{x} = \text{vec}(\{\mathbf{x}_v\})$	$\mathbf{x} = \text{bdiag}(\{\mathbf{X}_v\})$
$\tilde{\mathbf{x}}$, $\tilde{\mathbf{X}}$	LMC variable	$\tilde{\mathbf{x}} = \tilde{\mathbf{B}}\mathbf{x}$	$\tilde{\mathbf{X}} = \tilde{\mathbf{B}}\mathbf{X}\tilde{\mathbf{B}}^T$

Table 5
Notation for densities.

Symbol	Definition
$p(\dots)$	Regular density
$q(\dots)$	Density provided by the variational approximation
$\hat{p}(\dots)$	Density provided by the LMC surrogate

Appendix B. Proof for the bound on LMC regret

Mercer's theorem establishes that a positive semi-definite kernel $K(x_i, x_j)$ possesses an eigenfunction expansion (30), where $\lambda_0 \geq \lambda_1 \geq \lambda_2 \geq \cdots \geq 0$ are the eigenvalues associated to a set of orthogonal eigenfunctions $\psi_h(.)$. This implies that the variance-covariance matrices produced by evaluating the kernel on the N input observations in $\mathbf{X}$ can be written as the limit of a sequence:

$$\mathbf{K}^{\mathbf{x},\mathbf{x}} = \lim_{H\to\infty} \mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H (\mathbf{\Psi}_H^{\mathbf{x}})^T \tag{30}$$

$\mathbf{\Psi}_H^{\mathbf{x}}$ is an $N \times H$ matrix where the columns are the values produced by the h^{th} eigenfunction on the N data points and $\mathbf{\Lambda}_H$ is an $H \times H$ diagonal matrix containing the eigenvalues λ_h. The kernel is assumed to be a Hilbert-Schmidt operator, therefore $\sum_h \lambda_h^2 < \infty$, ensuring convergence of (30). Let $\mathcal{H}$ be the Hilbert space of all functions $\mathbf{f}^{\mathbf{x}}$ defined as linear combinations of these eigenfunctions using $1 \times H$ weights $\mathbf{f}_H$:

$$\mathbf{f}^{\mathbf{x}} = \lim_{H\to\infty} \mathbf{f}_H (\mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{\frac{1}{2}})^T = \lim_{H\to\infty} \mathbf{f}_H (\mathbf{K}^{\mathbf{x}})^T \tag{31}$$

Given a second function $\mathbf{g}^{\mathbf{x}} \in \mathcal{H}$, the inner product of this Hilbert space is defined as $\langle \mathbf{f}, \mathbf{g} \rangle_{\mathcal{H}} = \lim_{H\to\infty} \mathbf{f}_H \mathbf{g}_H^T$. From this it is possible to see that $\mathcal{H}$ meets the two criteria required of a reproducing kernel Hilbert space (RKHS):

$$\begin{cases} \langle \mathbf{K}^{\mathbf{x}}, \mathbf{K}^{\mathbf{x}} \rangle_{\mathcal{H}} = \lim_{H\to\infty} \mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{\frac{1}{2}} (\mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{\frac{1}{2}})^T = \mathbf{K}^{\mathbf{x},\mathbf{x}} \\ \langle \mathbf{f}_H, \mathbf{K}^{\mathbf{x}} \rangle_{\mathcal{H}} = \lim_{H\to\infty} \mathbf{f}_H (\mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{\frac{1}{2}})^T = \mathbf{f}^{\mathbf{x}} \end{cases} \tag{32}$$

The relevance of the RKHS to GP regression is that the predicted mean function of the LMC surrogate $\tilde{\boldsymbol{\mu}}^*$ (10) for an empirical input $\mathbf{x}^*$ is a linear combination of the kernel eigenfunctions (31) and is therefore in the RKHS. The eigenexpansion (30) therefore underpins the result of Micchelli et al. (2006) that the RKHS is universal if the Kernel is universal, and the following convergence result of Seeger et al. (2008), which bounds the regret of a kernel:

Lemma 1. *(from Seeger et al., 2008) Let K be a kernel possessing an eigenexpansion (30) and $\mathbf{K}^{\mathbf{x},\mathbf{x}}$ be the covariance matrix resulting from applying the kernel to a dataset $\mathbf{X}$ containing N observations drawn from density function $\nu(x)$. Then, given a constant $c > 0$:*

$$E_{\nu(x)}[R] \quad = \quad E_{\nu(x)}\left[\ln \left| \mathbf{I}_N + c\mathbf{K}^{\mathbf{x},\mathbf{x}} \right| \right] \quad \leq \quad \sum_{h=0}^{\infty} \ln\left(1 + c\lambda_h N\right)$$

Proof. The expected regret term $E_{\nu(x)}\left[\ln \left| \mathbf{I}_N + c\mathbf{K}^{\mathbf{x},\mathbf{x}} \right| \right]$ can be written in terms of the eigenexpansion (30) and rearranged using Sylvester's determinant identity:

$$\begin{aligned} E_{\nu(x)}\left[\ln \left| \mathbf{I}_N + c\mathbf{K}^{\mathbf{x},\mathbf{x}} \right| \right] &= \lim_{H\to\infty} E_{\nu(x)}\left[\ln \left| \mathbf{I}_N + c\mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H (\mathbf{\Psi}_H^{\mathbf{x}})^T \right| \right] \\ &= \lim_{H\to\infty} E_{\nu(x)}\left[\ln \left| \mathbf{I}_H + c\mathbf{\Lambda}_H^{1/2} (\mathbf{\Psi}_H^{\mathbf{x}})^T \mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{1/2} \right| \right] \end{aligned}$$

The expected regret can now be bounded using Jensen's inequality and the concavity of the logarithm:

$$\begin{aligned} E_{\nu(x)}\left[\ln \left| \mathbf{I}_N + c\mathbf{K}^{\mathbf{x},\mathbf{x}} \right| \right] &\leq \lim_{H\to\infty} \ln \left| \mathbf{I}_H + c E_{\nu(x)}\left[\mathbf{\Lambda}_H^{1/2} (\mathbf{\Psi}_H^{\mathbf{x}})^T \mathbf{\Psi}_H^{\mathbf{x}} \mathbf{\Lambda}_H^{1/2} \right] \right| \\ &= \lim_{H\to\infty} \ln \left| \mathbf{I}_H + c\mathbf{\Lambda}_H N E_{\nu(x)}\left[N^{-1} (\mathbf{\Psi}_H^{\mathbf{x}})^T \mathbf{\Psi}_H^{\mathbf{x}} \right] \right| \\ &= \sum_{h=0}^{\infty} \ln\left(1 + c\lambda_h N\right) \end{aligned}$$

The final expression for the bound relies on the fact that the orthogonality of the kernel eigenfunctions ensures that $E\left[N^{-1} (\mathbf{\Psi}_H^{\mathbf{x}})^T \mathbf{\Psi}_H^{\mathbf{x}} \right] = \mathbf{I}_H$. The resulting bound is thus the log determinant of a diagonal matrix, which is just the sum of the logarithm of the diagonal entries. □

In Seeger et al. (2008) this lemma directly provides the proof for the bound on regret. Changing the order of summation in the determinant using Sylvester's identity provides a bound expressed as an infinite sum over the eigenvalue spectrum of the kernel. This strategy does not work directly for the LMC kernel as its eigenexpansion (30) cannot be determined from the spectrum of the individual kernels in the sum. The following lemma shows, however, that the regret of the LMC kernel is bounded below a fixed multiple of the regret of worst-performing kernel in the linear combination:

Lemma 2. *Let $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}} = \sum_{v\in\mathbf{v}} \alpha_v \mathbf{K}_v^{\mathbf{x},\mathbf{x}}$ be a linear combination of V distinct $N \times N$ kernel matrices $\mathbf{K}_v^{\mathbf{x},\mathbf{x}}$ with weights $\alpha_v > 0$, where $\mathbf{v} = \{1, 2, ..., V\}$ is the set of indices identifying each kernel matrix and weight. Given $c > 0$, the following bound exists:*

$$\left|\mathbf{I}_N + c\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}\right| \quad < \quad \left|\mathbf{I}_N + Vc\alpha_{v^*}\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}\right|^V$$

Where:

$$v^* = \arg\max_{v\in\mathbf{v}} \left|\mathbf{I}_N + Vc\alpha_v\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\right|$$

Proof. As $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}$ is a linear combination of V kernels, we can write:

$$\left|\mathbf{I}_N + c\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}\right| = \left|\sum_{v\in\mathbf{v}} \mathbf{M}_v\right|, \qquad \mathbf{M}_v = \frac{1}{V}\mathbf{I}_N + c\alpha_v\mathbf{K}_v^{\mathbf{x},\mathbf{x}}$$

Factorizing out the determinants of the individual elements $|\mathbf{M}_v|$:

$$\begin{aligned}\left|\mathbf{I}_N + c\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}\right| &= \left(\prod_{v\in\mathbf{v}} |\mathbf{M}_v|\right)\left|\sum_{v\in\mathbf{v}}\prod_{p\in\mathbf{v}\setminus v} (\mathbf{M}_p)^{-1}\right| \\ &< \left(\prod_{v\in\mathbf{v}} |\mathbf{M}_v|\right)\left|\sum_{v\in\mathbf{v}}\prod_{p\in\mathbf{v}\setminus v} \frac{1}{\lambda_p^{min}}\mathbf{I}_N\right| \\ &\leq \left(\prod_{v\in\mathbf{v}} |\mathbf{M}_v|\right)|V\mathbf{I}_N|^V \quad = \quad \prod_{v\in\mathbf{v}}\left|\mathbf{I}_N + Vc\alpha_v\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\right|\end{aligned}$$

In the first bound λ_v^{min} is the smallest eigenvalue of $\mathbf{M}_v$. $\mathbf{K}_v^{\mathbf{x},\mathbf{x}}$ is positive semi-definite, therefore the eigenvalues of $\mathbf{M}_v$ are real-valued and $\lambda_v^{min} \geq 1/V$, which provides the second bound. Finally, if $v^* = \arg\max_{v\in\mathbf{v}} |\mathbf{M}_v|$, then:

$$\left|\mathbf{I}_N + c\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}\right| \quad < \quad \prod_{v\in\mathbf{v}}\left|\mathbf{I}_N + Vc\alpha_v\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\right| \quad < \quad \left|\mathbf{I}_N + Vc\alpha_{v^*}\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}\right|^V \quad \square$$

Lemmas 1 and 2 are sufficient to prove the bound on the expected regret of the LMC.

Proposition 1. *The expected regret of the LMC has the following upper bound, where v^* indicates the latent GP variable possessing the largest regret:*

$$E_{v(x)}\left[\ln\left|\mathbf{I}_{MN} + \boldsymbol{\Sigma}^{-2}\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}\right|\right] \quad < \quad MV\sum_{h=0}^{\infty}\ln\left(1 + Vb_{v^*,m^*}^2\sigma_{m^*}^{-2}\lambda_{v^*,h}N\right)$$

Proof. Rearranging the regret explicitly reveals the block diagonal structure of $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}$, induced by the use of the Kronecker product on the latent $\mathbf{K}_v^{\mathbf{x},\mathbf{x}}$:

$$R = \ln\left|\mathbf{I}_{MN} + \boldsymbol{\Sigma}^{-2}\tilde{\mathbf{B}}\mathbf{K}^{\mathbf{x},\mathbf{x}}\tilde{\mathbf{B}}^T\right| = \ln\left|\mathbf{I}_{MN} + \boldsymbol{\Sigma}^{-2}\left(\sum\nolimits_v \mathbf{B}_v\mathbf{B}_v^T \otimes \mathbf{K}_v^{\mathbf{x},\mathbf{x}}\right)\right|$$

This is bounded by the following sequence:

$$\begin{aligned}R &\leq \sum_m \ln\left|\mathbf{I}_N + \sigma_m^{-2}\sum_v b_{v,m}^2\mathbf{K}_v^{\mathbf{x},\mathbf{x}}\right| \\ &< \sum_m V\ln\left|\mathbf{I}_N + V\sigma_m^{-2}b_{v^*,m}^2\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}\right| \\ &< MV\ln\left|\mathbf{I}_N + V\sigma_{m*}^{-2}b_{v^*,m^*}^2\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}\right|\end{aligned}$$

First, the block diagonal structure of $\tilde{\mathbf{K}}^{\mathbf{x},\mathbf{x}}$ implies that Fischer's determinant inequality bounds R below the sum of the log determinants of the M main diagonal blocks, each a linear combination of V kernel matrices. In the second term, Lemma 2 bounds each log determinant by a multiple of the regret of the worst-performing kernel v^* for that variable m. Because v^* can differ across the M variables, the third bound is obtained by identifying the variable m^* with the largest regret term. At this point, R is bounded by an expression containing the regret of a single kernel matrix $\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}$. Lemma 1 provides the following bound on the expectation of that regret, completing the proof:

$$MV \times E_{v(x)}\left[\ln\left|\mathbf{I}_N + V\sigma_{m*}^{-2}b_{v^*,m^*}^2\mathbf{K}_{v^*}^{\mathbf{x},\mathbf{x}}\right|\right] \quad < \quad MV\sum_{h=0}^{\infty}\ln\left(1 + Vb_{v^*,m^*}^2\sigma_{m*}^{-2}\lambda_{v^*,h}N\right) \quad \square$$

Appendix. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2024.107972.

References

Alvarez, Mauricio A., Rosasco, Lorenzo, Lawrence, Neil D., et al., 2012. Kernels for vector-valued functions: a review. Found. Trends Mach. Learn. 4, 195–266.

Aushev, Alexander, Tran, Thong, Pesonen, Henri, Howes, Andrew, Kaski, Samuel, 2024. Likelihood-free inference in state-space models with unknown dynamics. Stat. Comput. 34, 27.
Barde, Sylvain, 2017. A practical, accurate, information criterion for nth order Markov processes. Comput. Econ. 50, 281–324.
Barde, Sylvain, 2020. Macroeconomic simulation comparison with a multivariate extension of the Markov information criterion. J. Econ. Dyn. Control 111.
Bargigli, Leonardo, Riccetti, Luca, Russo, Alberto, Gallegati, Mauro, 2020. Network calibration and metamodeling of a financial accelerator agent based model. J. Econ. Interact. Coord. 15, 413–440.
Bishop, Christopher M., 2006. Pattern Recognition and Machine Learning. Springer.
Burt, David, Rasmussen, Carl Edward, Van Der Wilk, Mark, 2019. Rates of convergence for sparse variational Gaussian process regression. In: International Conference on Machine Learning. PMLR, pp. 862–871.
Burt, David R., Rasmussen, Carl Edward, van der Wilk, Mark, 2020. Convergence of sparse variational inference in Gaussian processes regression. J. Mach. Learn. Res. 21, 1–63.
Caiani, Alessandro, Godin, Antoine, Caverzasi, Eugenio, Gallegati, Mauro, Kinsella, Stephen, Stiglitz, Joseph E., 2016. Agent based-stock flow consistent macroeconomics: towards a benchmark model. J. Econ. Dyn. Control 69, 375–408.
Caiani, Alessandro, Catullo, Ermanno, Gallegati, Mauro, 2018. The effects of fiscal targets in a monetary union: a multi-country agent-based stock flow consistent model. Ind. Corp. Change 27, 1123–1154.
Caiani, Alessandro, Russo, Alberto, Gallegati, Mauro, 2019. Does inequality hamper innovation and growth? An AB-SFC analysis. J. Evol. Econ. 29, 177–228.
Caponnetto, Andrea, Micchelli, Charles A., Pontil, Massimiliano, Ying, Yiming, 2008. Universal multi-task kernels. J. Mach. Learn. Res. 9, 1615–1646.
Chen, Siyan, Desiderio, Saul, 2022. A regression-based calibration method for agent-based models. Comput. Econ. 59, 687–700.
Choi, Taeryon, Schervish, Mark J., 2007. On posterior consistency in nonparametric regression problems. J. Multivar. Anal. 98, 1969–1987.
Cioppa, Thomas M., Lucas, Thomas W., 2007. Efficient nearly orthogonal and space-filling Latin hypercubes. Technometrics 49, 45–55.
Conti, Stefano, O'Hagan, Anthony, 2010. Bayesian emulation of complex multi-output and dynamic computer models. J. Stat. Plan. Inference 140, 640–651.
Cook, Samantha R., Gelman, Andrew, Rubin, Donald B., 2006. Validation of software for Bayesian models using posterior quantiles. J. Comput. Graph. Stat. 15, 675–692.
Delli Gatti, Domenico, Grazzini, Jakob, 2020. Rising to the challenge: Bayesian estimation and forecasting techniques for macroeconomic Agent Based Models. J. Econ. Behav. Organ. 178, 875–902.
Dias, Gustavo Fruet, Kapetanios, George, 2018. Estimation and forecasting in vector autoregressive moving average models for rich datasets. J. Econom. 202, 75–91.
Fagiolo, Giorgio, Moneta, Alessio, Windrum, Paul, 2007. A critical guide to empirical validation of agent-based models in economics: methodologies, procedures, and open problems. Comput. Econ. 30, 195–226.
Fagiolo, Giorgio, Guerini, Mattia, Lamperti, Francesco, Moneta, Alessio, Roventini, Andrea, 2019. Validation of agent-based models in economics and finance. In: Computer Simulation Validation. Springer, pp. 763–787.
Gardner, Jacob, Pleiss, Geoff, Weinberger, Kilian Q., Bindel, David, Wilson, Andrew G., 2018. Gpytorch: blackbox matrix-matrix Gaussian process inference with gpu acceleration. Adv. Neural Inf. Process. Syst. 31.
Gelman, Andrew, Roberts, Gareth O., Gilks, Walter R., 1996. Efficient Metropolis jumping rules. Bayesian Stat. 5, 42.
Gelman, Andrew, Gilks, Walter R., Roberts, Gareth O., 1997. Weak convergence and optimal scaling of random walk Metropolis algorithms. Ann. Appl. Probab. 7, 110–120.
Gilbert, Nigel, Troitzsch, Klaus, 2005. Simulation for the Social Scientist, 2nd edition. McGraw-Hill Education (UK).
Gilli, Manfred, Winker, Peter, 2003. A global optimization heuristic for estimating agent based models. Comput. Stat. Data Anal. 42, 299–312.
Gouriéroux, Christian, Monfort, Alain, 1993. Simulation based inference: a survey with special reference to panel data models. J. Econom. 59, 5–33.
Gouriéroux, Christian, Monfort, Alain, 1996. Simulation-Based Econometric Methods. Oxford University Press.
Gouriéroux, Christian, Monfort, Alain, Renault, Eric, 1993. Indirect inference. J. Appl. Econom. 8, S85–S118.
Grazzini, Jakob, Richiardi, Matteo, 2015. Estimation of ergodic agent-based models by simulated minimum distance. J. Econ. Dyn. Control 51, 148–165.
Grazzini, Jakob, Richiardi, Matteo G., Tsionas, Mike, 2017. Bayesian estimation of agent-based models. J. Econ. Dyn. Control 77, 26–47.
Gutmann, Michael U., Corander, Jukka, 2016. Bayesian optimization for likelihood-free inference of simulator-based statistical models. J. Mach. Learn. Res.
Hensman, James, Matthews, Alexander, Ghahramani, Zoubin, 2015. Scalable variational Gaussian process classification. In: Artificial Intelligence and Statistics. PMLR, pp. 351–360.
Hooten, Mevin, Wikle, Christopher, Schwob, Michael, 2020. Statistical implementations of agent-based demographic models. Int. Stat. Rev. 88, 441–461.
Hornik, Kurt, Stinchcombe, Maxwell, White, Halbert, 1989. Multilayer feedforward networks are universal approximators. Neural Netw. 2, 359–366.
Järvenpää, Marko, Gutmann, Michael U., Vehtari, Aki, Marttinen, Pekka, 2021. Parallel Gaussian Process Surrogate Bayesian Inference with Noisy Likelihood Evaluations.
Kennedy, Marc C., O'Hagan, Anthony, 2000. Predicting the output from a complex computer code when fast approximations are available. Biometrika 87, 1–13.
Kennedy, Marc C., O'Hagan, Anthony, 2001. Bayesian calibration of computer models. J. R. Stat. Soc., Ser. B, Stat. Methodol. 63, 425–464.
Koepernik, Peter, Pfaff, Florian, 2021. Consistency of Gaussian process regression in metric spaces. J. Mach. Learn. Res. 22, 1–27.
Kukačka, Jiří, Baruník, Jozef, 2017. Estimation of financial agent-based models with simulated maximum likelihood. J. Econ. Dyn. Control 85, 21–45.
Lamperti, Francesco, Roventini, Andrea, Sani, Amir, 2018. Agent-based model calibration using machine learning surrogates. J. Econ. Dyn. Control 90, 366–389.
Le Gratiet, Loic, Garnier, Josselin, 2015. Asymptotic analysis of the learning curve for Gaussian process regression. Mach. Learn. 98, 407–433.
Liefvendahl, Mattias, Stocki, Rafał, 2006. A study on algorithms for optimization of Latin hypercubes. J. Stat. Plan. Inference 136, 3231–3247.
Lu, Xuefei, Rudi, Alessandro, Borgonovo, Emanuele, Rosasco, Lorenzo, 2020. Faster kriging: facing high-dimensional simulators. Oper. Res. 68, 233–249.
Lueckmann, Jan-Matthis, Bassetto, Giacomo, Karaletsos, Theofanis, Macke, Jakob H., 2019. Likelihood-free inference with emulator networks. In: Symposium on Advances in Approximate Bayesian Inference. PMLR, pp. 32–53.
Lütkepohl, Helmut, 2005. New Introduction to Multiple Time Series Analysis. Springer Science & Business Media.
Meeds, Edward, Welling, Max, 2014. GPS-ABC: Gaussian process surrogate approximate Bayesian computation. arXiv preprint. arXiv:1401.2838.
Micchelli, Charles A., Xu, Yuesheng, Zhang, Haizhang, 2006. Universal kernels. J. Mach. Learn. Res. 7.
Neal, Radford M., 1996. Bayesian Learning for Neural Networks. Lecture Notes in Statistics, vol. 118. Springer Science & Business Media.
Papamakarios, George, Pavlakou, Theo, Murray, Iain, 2017. Masked autoregressive flow for density estimation. Adv. Neural Inf. Process. Syst. 30.
Papamakarios, George, Sterratt, David, Murray, Iain, 2019. Sequential neural likelihood: fast likelihood-free inference with autoregressive flows. In: The 22nd International Conference on Artificial Intelligence and Statistics. PMLR, pp. 837–848.
Platt, Donovan, 2021. Bayesian estimation of economic simulation models using neural networks. Comput. Econ., 1–52.
Rasmussen, Carl Edward, Williams, Christopher K., 2006. Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA.
Rudi, Alessandro, Camoriano, Raffaello, Rosasco, Lorenzo, 2015. Less is more: Nyström computational regularization. arXiv preprint. arXiv:1507.04717.
Salle, Isabelle, Yıldızoğlu, Murat, 2014. Efficient sampling and meta-modeling for computational economic models. Comput. Econ. 44, 507–536.
Santner, Thomas J., Williams, Brian J., Notz, William I., Williams, Brain J., 2018. The Design and Analysis of Computer Experiments, 2nd edition. Springer.
Schasfoort, Joeri, Godin, Antoine, Bezemer, Dirk, Caiani, Alessandro, Kinsella, Stephen, 2017. Monetary policy transmission in a macroeconomic agent-based model. Adv. Complex Syst. 20.
Seeger, Matthias W., Kakade, Sham M., Foster, Dean P., 2008. Information consistency of nonparametric Gaussian process methods. IEEE Trans. Inf. Theory 54, 2376–2382.

Shi, Jian Qing, Choi, Taeryon, 2011. Gaussian Process Regression Analysis for Functional Data. CRC Press.
Smets, Frank, Wouters, Rafael, 2007. Shocks and frictions in US business cycles: a Bayesian DSGE approach. Am. Econ. Rev. 97, 586–606.
Smith, Anthony A., 1993. Estimating nonlinear time-series models using simulated vector autoregressions. J. Appl. Econom. 8, S63–S84.
Smith, Anthony A., 2016. Indirect Inference. Palgrave Macmillan, UK.
Talts, Sean, Betancourt, Michael, Simpson, Daniel, Vehtari, Aki, Gelman, Andrew, 2018. Validating Bayesian inference algorithms with simulation-based calibration. arXiv preprint. arXiv:1804.06788.
Titsias, Michalis, 2009. Variational learning of inducing variables in sparse Gaussian processes. In: Artificial Intelligence and Statistics. PMLR, pp. 567–574.
Van Der Vaart, Aad, Van Zanten, Harry, 2011. Information rates of nonparametric Gaussian process methods. J. Mach. Learn. Res. 12.
Wood, Simon N., 2010. Statistical inference for noisy nonlinear ecological dynamic systems. Nature 466, 1102–1104.
Wynne, George, Briol, François-Xavier, Girolami, Mark, 2021. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. J. Mach. Learn. Res. 22.