Calibration and Uncertainty Quantification of Convective Parameters in an Idealized GCM

Oliver R. A. Dunbar[1], **Alfredo Garbuno-Inigo**[2], **Tapio Schneider**[1], **Andrew M. Stuart**[1]

[1]California Institute of Technology, Pasadena, California, USA
[2]Instituto Tecnológico Autónomo de México, Ciudad de México, México.

Key Points:

- We use time averaged climate statistics to calibrate convective parameters and quantify their uncertainties.
- We demonstrate use of the calibrate-emulate-sample algorithm to provide efficient calibration and uncertainty quantification.
- Parametric uncertainty in climate predictions is quantified by sampling from the learnt convective parameter distribution.

Corresponding author: Oliver Dunbar, odunbar@caltech.edu

Abstract

Parameters in climate models are usually calibrated manually, exploiting only small subsets of the available data. This precludes both optimal calibration and quantification of uncertainties. Traditional Bayesian calibration methods that allow uncertainty quantification are too expensive for climate models; they are also not robust in the presence of internal climate variability. For example, Markov chain Monte Carlo (MCMC) methods typically require $O(10^5)$ model runs and are sensitive to internal variability noise, rendering them infeasible for climate models. Here we demonstrate an approach to model calibration and uncertainty quantification that requires only $O(10^2)$ model runs and can accommodate internal climate variability. The approach consists of three stages: (i) a calibration stage uses variants of ensemble Kalman inversion to calibrate a model by minimizing mismatches between model and data statistics; (ii) an emulation stage emulates the parameter-to-data map with Gaussian processes (GP), using the model runs in the calibration stage for training; (iii) a sampling stage approximates the Bayesian posterior distributions by sampling the GP emulator with MCMC. We demonstrate the feasibility and computational efficiency of this calibrate-emulate-sample (CES) approach in a perfect-model setting. Using an idealized general circulation model, we estimate parameters in a simple convection scheme from synthetic data generated with the model. The CES approach generates probability distributions of the parameters that are good approximations of the Bayesian posteriors, at a fraction of the computational cost usually required to obtain them. Sampling from this approximate posterior allows the generation of climate predictions with quantified parametric uncertainties.

Plain Language Summary

Calibrating climate models with available data and quantifying their uncertainties are essential to make climate predictions accurate and actionable. A primary source of uncertainties in climate models comes from representation of small-scale processes such as moist convection. Parameters in convection schemes and other parameterizations are usually calibrated by hand, using only a small fraction of data that are available. As a result, the calibration process may miss information about the small-scale processes in question. This paper presents a proof-of-concept, in an idealized setting, of how parameters in climate models can be calibrated using a substantial fraction of the available data, and how uncertainties in the parameters can be quantified. We employ a new algorithm, called calibrate-emulate-sample (CES), which makes such calibration and uncertainty quantification feasible for computationally expensive climate models. CES reduces the hundreds of thousands of model runs usually required to quantify uncertainties in computer models to hundreds, thereby achieving about a factor 1000 speedup. It leads to more robust calibration and uncertainty quantification in the presence of noise arising from chaotic variability of the climate system. We show how uncertainties in climate model parameters can be translated into quantified uncertainties of climate predictions through ensemble integrations.

1 Introduction

The principal uncertainties in climate predictions arise from the representation of unresolvable yet important small-scale processes, such as those controlling cloud cover (Cess et al., 1989, 1990; Bony & Dufresne, 2005; Stephens, 2005; Bony et al., 2006; Vial et al., 2013; Webb et al., 2013; Brient & Schneider, 2016; Schneider, Teixeira, et al., 2017). These processes are represented by parameterization schemes, which relate unresolved quantities such as cloud statistics to variables resolved on the climate models' computational grid, such as temperature and humidity. The parameterization schemes depend on parameters that are a priori unknown, and so fixing the parameters is associated with uncertainty. The process of fixing these parameters to values that are most consistent

with data is known as calibration, which generally requires solving an optimization problem. Traditionally, however, parameters are calibrated ("tuned") by hand, in a process that exploits only a small subset of the available observational data and relies on the knowledge and intuition of climate modelers about plausible ranges of parameters and their effect on the simulated climate of a model (Randall & Wielicki, 1997; Mauritsen et al., 2012; Golaz et al., 2013; Hourdin et al., 2013; Flato et al., 2013; Hourdin et al., 2017; Schmidt et al., 2017; Zhao et al., 2018). More recently, some broader-scale automated approaches that more systematically quantify the plausible range of parameters have begun to be explored (Couvreux et al., 2020; Hourdin et al., 2020).

Opportunities to improve climate models lie in exploiting a larger fraction of the available observational data together with high-resolution simulations, and learning from both systematically and not manually (Schneider, Lan, et al., 2017). To fully account for parametric uncertainty, we adopt a Bayesian view of the model-data relationship, which amounts to solving a Bayesian inverse problem. Given model, data, and prior information on the parameters, Bayesian inversion yields a posterior distribution of parameters. The mean or mode of the posterior distribution define the best parameter estimate, and the entire distribution provides uncertainty quantification through its spread about the mean or mode. We use the mismatch between climate statistics simulated with the model and those obtained from observations or high-resolution simulations as the data likelihood in the Bayesian inverse problem to calibrate parameterizations in a climate model and to quantify their parametric uncertainties. Our focus is on learning from time-averaged climate statistics for three reasons: (1) time-averaged statistics are what is relevant for climate predictions; (2) time-averaged statistics vary more smoothly in space than atmospheric states, leading to a smoother optimization problem than that of atmospheric state estimation in numerical weather prediction (NWP); (3) time-averaging over long time-intervals reduces the effect of the unknown initial state of the system, removing the need to determine it. Focusing on time-averaged climate statistics, rather than on instantaneous states or trajectories as in NWP, makes it possible to exploit climate observations and high-resolution simulations even when their native resolutions are very different from those of climate models.

While learning from climate statistics accumulated in time presents opportunities, it also comes with challenges. Accumulating statistics in time is computationally much more expensive than the forecasts over hours or days used in NWP. Therefore, we need algorithms for learning from data that minimize the number of climate model runs required. Traditional methods for Bayesian calibration and uncertainty quantification such as Markov chain Monte Carlo (MCMC) typically require many iterations—often more than 10^5—to reach statistical convergence (e.g., Geyer, 2011). Conducting so many computationally expensive climate model runs is not feasible, rendering MCMC impractical for climate model calibration (Annan & Hargreaves, 2007). Additionally, while MCMC can be used to obtain the distribution of model parameters given data, it is not robust with respect to noise in the evaluation of the map from model parameters to data. Such noise, arising from natural variability in the chaotic climate system, can lead to trapping of the Markov chains in spurious, noise-induced local maxima of the likelihood function (Cleary et al., 2021). This presents additional challenges to using MCMC methods for climate model calibration.

Here we demonstrate a new approach to climate model uncertainty quantification that overcomes the limitations of traditional Bayesian calibration methods in a relatively simple proof-of-concept. The approach—called calibrate-emulate-sample (CES) (Cleary et al., 2021)—consists of three successive stages, which each exploit proven concepts and methods:

1. In a calibration stage, we use variants of ensemble Kalman methods. These approaches were originally developed as a derivative-free method for state estima-

tion (Evensen, 1994; Van Leeuwen & Evensen, 1996) and are now widely used in NWP (Houtekamer & Zhang, 2016). The methods were subsequently developed for simultaneous state and parameter estimation, and variants were developed to deal with strongly nonlinear systems (Bocquet & Sakov, 2012; Gu & Oliver, 2007; Li & Reynolds, 2009; Sakov et al., 2012; Bocquet & Sakov, 2014). They were eventually recognized as a general purpose tool for the solution of inverse problems whose objective is parameter estimation (Chen & Oliver, 2012; Emerick & Reynolds, 2013; Reich, 2011; Evensen, 2018). Here we use an optimization approach, referred to as ensemble Kalman inversion (Iglesias et al., 2013), which builds on the work of (Chen & Oliver, 2012; Emerick & Reynolds, 2013). However, ensemble Kalman inversion and other ensemble Kalman methods do not provide a basis for systematic uncertainty quantification, except in linear Gaussian problems (Annan & Hargreaves, 2007; Gland et al., 2009; Ernst et al., 2015).

2. In an emulation stage, we train an emulator on the climate model statistics generated during the calibration stage in order to quantify uncertainties. To emulate how the climate model statistics depend on parameters to be calibrated, we use Gaussian processes (GPs), a hierarchical machine learning method that learns smooth functions from a set of noisy training points (Kennedy & O'Hagan, 2001; Santner et al., 2018). The GP approach also learns the statistics of the uncertainty in the resulting predicted function. The training points here are provided by the climate model runs performed in the calibration stage.
3. In a sampling stage, we approximate the posterior distribution on parameters given model and data, using the GP emulator to replace the parameter-to-climate statistics map. The emulator is used to estimate error statistics and to sample from the approximate posterior distribution with MCMC. Because the GP emulator is computationally cheap to evaluate and is smooth by virtue of the smoothing properties of GPs, this avoids the issues that limit the usability of MCMC for sampling from climate models directly.

The CES approach is described in detail in Cleary et al. (2021), which provides a justification and contextualization of the approach in the literature on data assimilation and Bayesian calibration. If uncertainty quantification is not required, the calibration step provides an effective, derivative-free parameter estimation method, and the emulation and sampling stages are not required. However, when uncertainty quantification is required, the role of the calibration stage is to provide a good set of training points in the vicinity of the posterior mean or mode, to train the emulator in the next stage of the algorithm.

The purpose of this paper is to demonstrate the feasibility of the approach for estimating parameters in an idealized general circulation model (GCM). This represents a proof-of-concept in a small parameter space and limited data space; how the methods scale up to larger problems will be discussed at the end.

This paper is arranged as follows: Section 2 describes the experimental setup, including the idealized GCM and the generation of synthetic data from it. Section 3 describes the CES approach and the methods used in each stage. Section 4 describes the results of numerical experiments that use CES to calibrate parameters in the idealized GCM and quantify their uncertainties. It also demonstrates how sampling from the posterior distribution of parameters can be used to generate climate predictions with quantified uncertainties. Section 5 discusses and summarizes the results and their applicability to larger problems.

2 Experimental Setup

2.1 General Circulation Model

We use the idealized GCM described by Frierson et al. (2006) and O'Gorman and Schneider (2008b), which is based on the spectral dynamical core of the Flexible Modeling System developed at the Geophysical Fluid Dynamics Laboratory. To approximate the solution of the hydrostatic primitive equations, it uses the spectral transform method in the horizontal, with spectral resolution T21 and with 32 latitude points on the transform grid. It uses finite differences with 10 unevenly spaced sigma levels in the vertical. We chose this relatively coarse resolution to keep our numerical experiments computationally efficient, so that comparison of CES with much more expensive methods is feasible. The lower boundary of the GCM is a homogeneous slab ocean (1 m mixed-layer thickness). Radiative transfer is represented by a semi-gray, two-stream radiative transfer scheme, in which the optical depth of longwave and shortwave absorbers is a prescribed function of latitude and pressure (O'Gorman & Schneider, 2008b), irrespective of the concentration of water vapor in the atmosphere (i.e., without an explicit representation of water vapor feedback). Insolation is constant and approximates Earth's annual mean insolation at the top of the atmosphere.

We focus our calibration and uncertainty quantification experiments on parameters in the GCM's convection scheme, which is a quasi-equilibrium moist convection scheme that can be viewed as a simplified version of the Betts-Miller convection scheme (Betts, 1986; Betts & Miller, 1986, 1993). It relaxes temperature T and specific humidity q toward reference profiles on a timescale τ (Frierson, 2007):

$$\frac{\partial T}{\partial t} + \cdots = -f_T \frac{T - T_{\mathrm{ref}}}{\tau} \tag{1}$$

and

$$\frac{\partial q}{\partial t} + \cdots = -f_T f_q \frac{q - q_{\mathrm{ref}}}{\tau}. \tag{2}$$

Here, $f_T(z; T, q, p)$ is a function of altitude z and of the thermodynamic state of an atmospheric column (dependent on temperature T, pressure p, and specific humidity q in the column), which determines where and when the convection scheme is active; $f_q(T, q, p)$ is a function that modulates the relaxation of the specific humidity in non-precipitating (shallow) convection (Frierson, 2007; O'Gorman & Schneider, 2008b). The reference temperature profile is a moist adiabat, $T_{\mathrm{ma}}(z)$, shifted by a state-dependent and constant-with-height offset ΔT, which is chosen to ensure conservation of enthalpy integrated over a column: $T_{\mathrm{ref}}(z) = T_{\mathrm{ma}}(z) + \Delta T$. The reference specific humidity $q_{\mathrm{ref}}(z)$ is the specific humidity corresponding to a fixed relative humidity RH relative to the moist adiabat $T_{\mathrm{ma}}(z)$. The two key parameters in this simple convection scheme thus are the timescale τ and the relative humidity RH. We demonstrate how we can learn about them from synthetic data generated with the GCM.

2.2 Variable Selection and Generation of Synthetic Data

The idealized GCM with the simple quasi-equilibrium convection scheme has been used in numerous studies of large-scale atmosphere dynamics and mechanisms of climate changes, especially those involving the hydrologic cycle (e.g., O'Gorman & Schneider, 2008b, 2008a; Bordoni & Schneider, 2008; O'Gorman & Schneider, 2009b; Schneider et al., 2010; Merlis & Schneider, 2011; O'Gorman, 2011; Kaspi & Schneider, 2011, 2013; Levine & Schneider, 2015; Bischoff & Schneider, 2014; Wills et al., 2017; Wei & Bordoni, 2018). We know from this body of work that the convection scheme primarily affects the atmospheric thermal stratification in the tropics, with weaker effects in the extratropics (Schneider & O'Gorman, 2008). We also know that the relative humidity parameter (RH) in the moist convection scheme controls the humidity of the tropical free troposphere but likewise has a weaker effect on the humidity of the extratropical free troposphere (O'Gorman

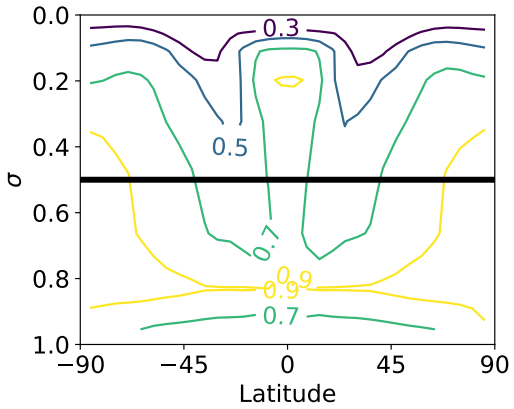

Figure 1. Zonal average of relative humidity averaged over one month. The black line shows the level at which data was extracted for computing data misfit functions.

et al., 2011). Thus, we expect tropical circulation statistics to be especially informative about the parameters in the convection scheme. However, convection plays a central role in extreme precipitation events at all latitudes (O'Gorman & Schneider, 2009b, 2009a), so we expect statistics of precipitation extremes to be informative about convective parameters, and in particular to contain information about the relaxation timescale τ.

As climate statistics from which we want to learn about the convective parameters, we choose 30-day averages of the free-tropospheric relative humidity, of the precipitation rate, and of a measure of the frequency of extreme precipitation. Because the GCM is statistically zonally symmetric, we take zonal averages in addition to the time averages. The relative humidity is evaluated at $\sigma = 0.5$ (where $\sigma = p/p_s$ is pressure p normalized by the local surface pressure p_s), as shown in Figure 1. As a measure of the frequency of precipitation extremes, we use the probability that daily precipitation rates exceed a high, latitude-dependent threshold. The threshold is chosen as the latitude-dependent 90th percentile of daily precipitation in a long (18000 days) control simulation of the GCM in a statistically steady state. So for the parameters in the control simulation, the precipitation threshold is expected to be exceeded 10% of the time at each latitude. The convective parameters in the control simulation are fixed at their reference values RH = 0.7 and $\tau = 2$ h (O'Gorman & Schneider, 2008b), and we collect the parameters in the vector $\boldsymbol{\theta}^\dagger = (\theta^\dagger_{\mathrm{RH}}, \theta^\dagger_\tau) = (0.7, 2\text{ h})$. Figure 2 shows the mean relative humidity, the mean precipitation rate (broken down into its contributions coming from the convection scheme and from condensation at resolved scales), and the 90th percentile precipitation rate, from the control simulation averaged over 600 batches of 30-day windows. We use the single long control simulations of duration 18000 days only for the creation of Figure 2 and for the estimation of noise covariances.

2.3 Definition of noise covariance

Estimation of model parameters requires specification of a noise covariance matrix, reflecting errors and uncertainties in the data. The principal source of noise in our perfect-model setting with synthetic data is sampling variability due to finite-time averaging with unknown initial conditions. The initial condition is forgotten at sufficiently long times because of the chaotic nature of atmospheric variability, so a central limit theorem quantifies the finite-time fluctuations around infinite-time averages that are caused by uncertain initial conditions. Therefore, the asymptotic distribution of the fluctuations is a multivariate normal distribution $N(0, \Sigma(\boldsymbol{\theta}))$ with zero mean and covariance matrix $\Sigma(\boldsymbol{\theta})$. We

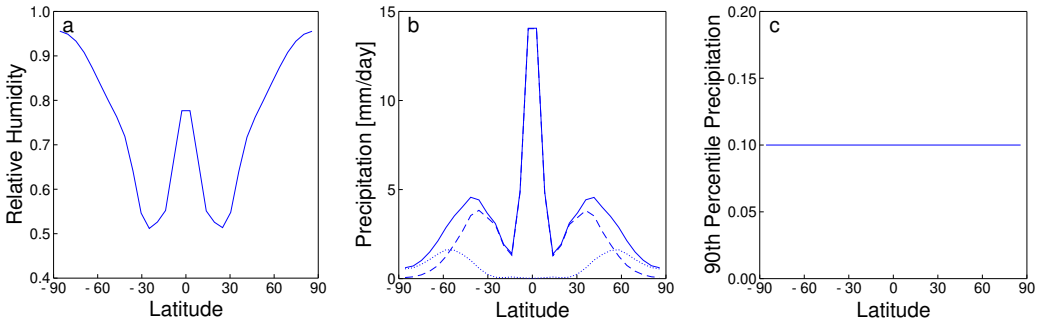

Figure 2. Long-term mean values of synthetic data. (a) Free-tropospheric relative humidity. (b) Total daily precipitation rate (solid) and its contributions from convection (dashed) and grid-scale condensation (dotted). (c) Probability of daily precipitation exceeding a 90th percentile (which is trivially 10% in this case).

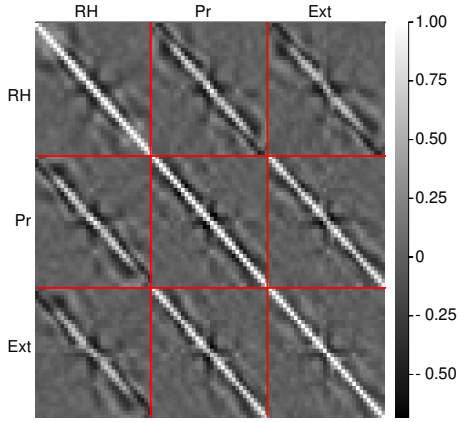

Figure 3. Correlation matrix associated with the internal variability estimated from the control simulation, plotted to illustrate the structure of Σ. The matrix blocks labeled (RH, Pr, Ext) are associated with the observed relative humidity, precipitation, and extreme precipitation.

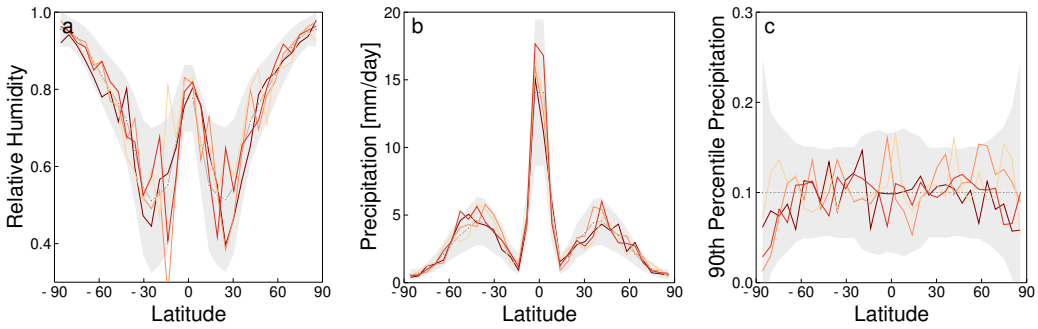

Figure 4. Four noisy realizations of the synthetic data, plotted in color over the underlying mean (grey circles) and 95% confidence intervals from $\Gamma(\theta^\dagger)$ (grey bars). (a) Relative humidity. (b) Daily precipitation rate. (c) Probability of daily precipitation exceeding the 90th percentile of the long-term mean data.

estimate the covariance matrix at $\Sigma(\boldsymbol{\theta}^\dagger)$, that is, with the parameters $\boldsymbol{\theta}^\dagger$ in the control simulation. To estimate $\Sigma(\boldsymbol{\theta}^\dagger)$, we run the GCM for 600 windows of length 30 days (because we use 30-day averages to estimate parameters) and calculate the sample covariance matrix of the 30-day means. With the 3 latitude-dependent fields evaluated at 32 latitude points, $\Sigma(\boldsymbol{\theta}^\dagger)$ is a 96×96 symmetric matrix representing noise from internal variability in finite-time averages. Hereafter, we make the assumption that $\Sigma(\boldsymbol{\theta}) \approx \Sigma(\boldsymbol{\theta}^\dagger)$ for any $\boldsymbol{\theta}$, and thus we treat Σ as a constant matrix. In practical implementations of this method, a corresponding constant Σ can be estimated from climatology. We illustrate the correlation structure of Σ in Figure 3.

To generate synthetic data, we also include the effect of measurement error (Kennedy & O'Hagan, 2001). We add Gaussian noise to the time-averaged model statistics, with a diagonal covariance structure in data space. We construct the measurement error covariance matrix Δ to be diagonal with entries $\delta_i > 0$, where i indexes over data type (the 3 observed quantities) and latitude (32 locations). Combining this measurement covariance matrix Δ with the covariance matrix Σ arising from internal variability leads to an inflated noise covariance matrix

$$\Gamma = \Sigma + \mathrm{diag}(\delta_i^2) = \Sigma + \Delta. \tag{3}$$

There are many options to pick δ_i. We choose it by reducing the distance of the 95% confidence interval to its nearest physical boundary for each i by a constant factor C, so as to retain physical properties (e.g., precipitation must be nonnegative). Denote the mean μ_i, variance Σ_{ii}, and a physical boundary set $\partial\Omega_i$ for each data i, we choose

$$\delta_i = C \min\left(\mathrm{dist}(\mu_i + 2\sqrt{\Sigma_{ii}}, \partial\Omega_i), \mathrm{dist}(\mu_i - 2\sqrt{\Sigma_{ii}}, \partial\Omega_i)\right).$$

We take $C = 0.2$. This value implies a significant noise inflation, with an average ratio of the standard deviations $\sqrt{\Gamma_{ii}}/\sqrt{\Sigma_{ii}}$ of 2.3. Figure 4 shows the resulting data mean (grey circles), the 95% confidence interval of the inflated covariance (grey ribbon), and four realizations of the data $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$ (yellow to red lines), each defined by taking a different 30-day average of the GCM, and adding a different realization of $N(0, \Delta)$. These four realizations will be used throughout when presenting our results.

3 Methods

3.1 Misfit functions for time averaged data

Both calibration and uncertainty quantification in CES rely on a misfit function (standardized error) that quantifies mismatch between model output and data. Calibration minimizes a (possibly regularized) misfit function over the parameter space; uncertainty quantification samples from the posterior distribution, using a misfit function as the negative log-likelihood. To define the desired misfit function, we introduce $\mathcal{G}_T(\boldsymbol{\theta}; \boldsymbol{z}^{(0)})$ and $\mathcal{G}_\infty(\boldsymbol{\theta})$, which denote the mapping from the parameter vector $\boldsymbol{\theta}$ to the 96 data points, either averaged over a finite time horizon (T) or over an infinite time horizon (∞). The former average depends on the unknown initial condition $\boldsymbol{z}^{(0)}$, whereas the latter does not, because the initial condition is forgotten after a sufficiently long time. We refer to $\mathcal{G}_T(\boldsymbol{\theta}; \boldsymbol{z}^{(0)})$ as the forward model and $\mathcal{G}_\infty(\boldsymbol{\theta})$ as the infinite time-horizon forward model.

To define the misfit function, we begin from the relationship between parameters $\boldsymbol{\theta}$ and data $\boldsymbol{y}$. Expressed in terms of finite-time averages, this relationship has the form

$$\boldsymbol{y} = \mathcal{G}_T(\boldsymbol{\theta}; \boldsymbol{z}^{(0)}) + \boldsymbol{\eta}, \qquad \boldsymbol{\eta} \sim N(0, \Delta). \tag{4}$$

The data realizations $\boldsymbol{y}^{(i)}$ for $i = 1, \ldots, 4$ in Section 2.3 can therefore be seen as evaluations of (4) for initial conditions $\boldsymbol{z}_i^{(0)}$, noise realization $\boldsymbol{\eta}_i \sim N(0, \Delta)$, and at a parameter $\boldsymbol{\theta}^\dagger$. This form has the undesirable feature that it involves $\boldsymbol{z}^{(0)}$, a quantity that is not of intrinsic interest. However, the central limit theorem asserts that

$$\mathcal{G}_T(\boldsymbol{\theta}; \boldsymbol{z}^{(0)}) \approx \mathcal{G}_\infty(\boldsymbol{\theta}) + \sigma,$$

where $\sigma \sim N(0, \Sigma)$. This theorem quantifies the forgetting of the initial condition after sufficiently long times, e.g., for the atmosphere, $T \gtrsim 15$ days (Zhang et al., 2019). From the definition (3) of the total noise covariance matrix, $\Gamma = \Sigma + \Delta$, we can combine the observational and internal-variability noise and write

$$\boldsymbol{y} = \mathcal{G}_\infty(\boldsymbol{\theta}) + \boldsymbol{\gamma}, \qquad \boldsymbol{\gamma} \sim N(0, \Gamma). \tag{5}$$

We can equally interpret $\boldsymbol{y}^{(i)}$ for $i = 1, \ldots, 4$ from Section 2.3 as evaluations of (5) with noise $\boldsymbol{\gamma}_i \sim N(0, \Gamma)$ and at parameter $\boldsymbol{\theta}^\dagger$. This removes the dependence on initial condition but is expressed in terms of infinite-time averages. Computing these averages directly is not feasible, but (in the emulation phase) we introduce a procedure that enables us to learn a surrogate model for $\mathcal{G}_\infty$, using carefully chosen finite-time model evaluations $\mathcal{G}_T$.

In the Bayesian approach to parameter learning, the aim is to determine the conditional distribution of parameters $\boldsymbol{\theta}$ given a realization of data $\boldsymbol{y}$ (written $\boldsymbol{\theta} \mid \boldsymbol{y}$) by applying Bayes theorem, which states $\mathbb{P}(\boldsymbol{\theta} \mid \boldsymbol{y})$ is proportional to the product of the likelihood $\mathbb{P}(\boldsymbol{y} \mid \boldsymbol{\theta})$ and the prior $\mathbb{P}(\boldsymbol{\theta})$. In the case where a surrogate model for $\mathcal{G}_\infty$ is available, $\boldsymbol{y} \mid \boldsymbol{\theta}$ is defined as the pushforward of $\boldsymbol{\theta}$ through a parameter-to-data map (5), and we define the misfit function, the negative logarithm of the likelihood, as

$$\Phi(\boldsymbol{\theta}, \boldsymbol{y}) = \frac{1}{2} \|\boldsymbol{y} - \mathcal{G}_\infty(\boldsymbol{\theta})\|_\Gamma^2 , \tag{6}$$

where $\|\cdot\|_\Gamma = \|\Gamma^{-1/2} \cdot\|_2$ is the Mahalanobis distance. Before a surrogate model for $\mathcal{G}_\infty$ is available, this function is infeasible to evaluate, but as $\boldsymbol{y} \mid \boldsymbol{\theta}$ can be defined using (4) as well, we consider the related misfit function

$$\Phi_T(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{z}^{(0)}) = \frac{1}{2} \|\boldsymbol{y} - \mathcal{G}_T(\boldsymbol{\theta}; \boldsymbol{z}^{(0)})\|_{\Gamma+\Sigma}^2 . \tag{7}$$

We view evaluation of $\mathcal{G}_T$ from any initial condition as a random approximation of $\mathcal{G}_\infty$; hence, the additional internal-variability covariance matrix Σ appears in (7).

The CES algorithm in this context proceeds as follows: use optimization based on (7) to calibrate parameters; on the basis of evaluations of $\mathcal{G}_T$ made during the calibration, learn a GP surrogate for $\mathcal{G}_\infty$; then use this surrogate to sample from the posterior distribution of $(\boldsymbol{\theta} \mid \boldsymbol{y})$ defined using (6). We will henceforth neglect $\boldsymbol{z}^{(0)}$ in our notation, and just write $\mathcal{G}_T(\boldsymbol{\theta})$ and $\Phi_T(\boldsymbol{\theta}, \boldsymbol{y})$. Viewing the initial condition as random makes these objects random as well.

We have the following undesirable properties of the finite-time model average $\mathcal{G}_T(\boldsymbol{\theta})$: (i) it is computationally expensive to evaluate for large T; (ii) it can be nondifferentiable or difficult to differentiate (e.g., because of non-differentiability of parameterization schemes in climate models); and (iii) evaluations of it are not deterministic (when one drops the explicit dependence on initial conditions). Our methodology, detailed in the upcoming sections, is constructed to overcome these difficulties.

An alternative approach to emulating the map $\mathcal{G}_\infty(\boldsymbol{\theta})$ is to emulate $\Phi(\boldsymbol{\theta}, \boldsymbol{y})$ directly. This is often more computationally efficient as one always models a scalar function. During investigation however, we found the efficiency comes at a cost of reduced accuracy and interpretability; in preliminary experiments these drawbacks proved detrimental to performance, and so we abandoned the approach.

3.2 Prior distributions

The priors of the physical parameters are taken to be the logit-normal and lognormal distributions, $\theta_{RH} \sim \text{Logitnormal}(0, 1)$ and $\theta_\tau \sim \text{Lognormal}(12\text{ h}, (12\text{ h})^2)$, for the relative humidity and timescale parameter, respectively. Define the invertible transformation $\mathcal{T} : \mathbb{R}^2 \to \mathbb{R}^2$ by $\mathcal{T}(x, y) = (\text{logit}(x), \log(y))$. Through an abuse of notation, we relabel the transformed (or computational) parameters as $\boldsymbol{\theta}$, and the untransformed (or physical) parameters (relative humidity and timescale) are uniquely defined by $\mathcal{T}^{-1}(\boldsymbol{\theta})$. The forward map $\mathcal{G}_T$ is defined as a composition $\mathcal{F}_T \circ \mathcal{T}^{-1}$ where $\mathcal{F}_T$ is the forward map defined on the physical parameters.

This choice allows us to apply our methods in the transformed space, where our priors are normally distributed and unbounded (namely $\text{logit}(\theta_{RH}) \sim N(0, 1)$ and $\log(\theta_\tau) \sim N(10.17, 1)$); meanwhile the climate model is applied only to physically defined variables, $\theta_{RH} \in [0, 1]$ and $\theta_\tau \in [0, \infty)$. In this way the prior distributions enforce physical constraints on the parameters.

3.3 Calibrate: Ensemble Kalman Inversion

Ensemble Kalman inversion (EKI) (Iglesias et al., 2013) is an offline variant of ensemble Kalman filtering designed to learn parameters in a general model, rather than states of a dynamical system. EKI can be viewed as a derivative-free optimization algorithm. Given a set of data $\boldsymbol{y}$, it iteratively evolves an ensemble of parameter estimates both so that they achieve consensus and evolve toward the optimal parameter value $\boldsymbol{\theta}^*$ (close to $\boldsymbol{\theta}^\dagger$ if the quality/volume of the data is good enough) that minimizes an objective function, in our case given by the misfit (7), possibly with inclusion of a regularization term. It has great potential for use with chaotic or stochastic models due to its ensemble-based, derivative-free approach for optimizing parameters. Theoretical work shows that noisy continuous-time versions of EKI exhibit an averaging effect that skips over fluctuations superimposed onto the ergodic averaged forward model (Duncan et al., 2021). Empirical results suggest similar phenomena apply to EKI as implemented here, justifying the application to noisy forward model evaluations. Furthermore, the derivative-free approach scales well to high-dimensional parameter spaces, as evidenced by the use of Kalman filtering in numerical weather prediction, where billions of parameters characterizing atmospheric states are routinely estimated (Kalnay, 2002). This makes the algorithm appealing for complex climate models. The algorithm is mathematically proven

to find the optimizer, within an initial, ensemble-dependent subspace, for linear models (Schillings & Stuart, 2017a). It is known to be effective for high-dimensional nonlinear models (Iglesias et al., 2013; Schneider et al., 2020c, 2020a), such as the nonlinear map from parameters to data represented by the idealized GCM we use in our proof-of-concept here.

The EKI algorithm we use is detailed in Iglesias et al. (2013). The algorithm iteratively updates an ensemble of parameters, $\boldsymbol{\theta}_m^{(n)}$, where $m = 1, \dots M$ denotes an ensemble member, and the superscript n denotes the iteration count. The algorithm uses the ensemble to update parameters according to the following equation

$$\boldsymbol{\theta}_m^{(n+1)} = \boldsymbol{\theta}_m^{(n)} + C_{\boldsymbol{\theta}\mathcal{G}}^{(n)} \left(\Gamma + C_{\mathcal{G}\mathcal{G}}^{(n)}\right)^{-1} \left(\boldsymbol{y} - \mathcal{G}_T(\boldsymbol{\theta}_m^{(n)})\right) ,$$

where $C_{\mathcal{G}\mathcal{G}}$ is the empirical covariance of the ensemble of quantities of interest from model runs, and $C_{\boldsymbol{\theta}\mathcal{G}}$ is the empirical cross-covariance of the ensemble of parameters and the ensemble of quantities of interest. The covariance matrix of the distribution of differences between realizations of $\boldsymbol{y}$ and $\mathcal{G}_T(\cdot)$ is $\Gamma+\Sigma$, which is approximated empirically in the update by $\Gamma+C_{\mathcal{G}\mathcal{G}}^{(n)}$. When ensemble methods are used as approximate samplers (Chen & Oliver, 2012; Emerick & Reynolds, 2013), additional independent noise is added to $\boldsymbol{y}$ at each iteration and for every ensemble member; however, because we are solving an optimization problem within this calibration phase, such noise is not added here.

We initialize the algorithm by drawing an ensemble of size $M = 100$ by sampling the parameter space from assumed prior distributions on the parameters.

3.4 Emulate: Gaussian Process Emulators (EKI-GP)

During the calibration stage with N iterations and an ensemble of size M, we obtain a collection of input–output pairs

$$\{\boldsymbol{\theta}_m^{(n)}, \mathcal{G}_T(\boldsymbol{\theta}_m^{(n)})\}, \qquad n = 0, \dots, N, \quad m = 1, \dots M.$$

The cloud of points $\{\boldsymbol{\theta}_m^{(n)}\}$ from the calibration stage (a) spans the prior distribution, as initial EKI draws are from the prior, and (b) in later iterations, has a high density around the point $\boldsymbol{\theta}^*$ to which EKI converges. We use regression to train a GP emulator mapping $\boldsymbol{\theta}$ to $\mathcal{G}_T(\boldsymbol{\theta})$, using the input–output pairs $\{\boldsymbol{\theta}_m^{(n)}, \mathcal{G}_T(\boldsymbol{\theta}_m^{(n)})\}$, which are referred to as training points in the context of GP regression. The emulation will be most accurate in regions with more training points, that is, around $\boldsymbol{\theta}^*$. This is typically near the true solution $\boldsymbol{\theta}^\dagger$, and it is the region where the posterior parameter distribution will have high probability; this is precisely where uncertainty quantification requires accuracy. In effect, EKI serves as an effective algorithm for selecting good training points for GP regression.

Gaussian processes emulate the statistics of the input–output pairs, using a Gaussian assumption. Specifically, we learn an approximation of the form

$$\mathcal{G}_T(\boldsymbol{\theta}) \approx \mathcal{N}(\mathcal{G}_{\mathrm{GP}}(\boldsymbol{\theta}), \Sigma_{\mathrm{GP}}(\boldsymbol{\theta})).$$

The approximation is learned from the input-output pairs assuming that the outputs are produced from a mean function $\mathcal{G}_{\mathrm{GP}}(\boldsymbol{\theta})$, and subject to normally distributed noise defined by a covariance function $\Sigma_{\mathrm{GP}}(\boldsymbol{\theta})$, both dependent on the parameters. The choice of notation here is to imply the fact that $\mathcal{G}_{\mathrm{GP}}(\boldsymbol{\theta})$ serves to approximate the (unattainable) infinite-time average of the model $\mathcal{G}_\infty(\boldsymbol{\theta})$, and $\Sigma_{\mathrm{GP}}(\boldsymbol{\theta})$ serves to approximate the covariance matrix Σ. Importantly, $\Sigma_{\mathrm{GP}}(\boldsymbol{\theta})$ is $\boldsymbol{\theta}$-dependent as it also includes the uncertainty in the approximation of the emulator at $\boldsymbol{\theta}$ (for example, the emulator uncertainty $\Sigma_{\mathrm{GP}}(\boldsymbol{\theta})$ will be large when $\boldsymbol{\theta}$ is far from the inputs $\{\boldsymbol{\theta}_m\}$ used in training).

The atmospheric quantities from which we learn about model parameters are correlated (e.g., relative humidity or daily precipitation at neighboring latitudes are correlated), resulting in a nondiagonal covariance matrix Σ. Any GP emulator therefore also

requires a nondiagonal covariance $\Sigma_{\text{GP}}(\boldsymbol{\theta})$. We can enforce this, by mapping the correlated statistics from the GCM into a decorrelated space by using a principal component analysis on Σ, and then training the GP with the decorrelated statistics to produce an emulator with diagonal covariance $\tilde{\Sigma}_{\text{GP}}(\boldsymbol{\theta})$. We use the notation $\tilde{(\cdot)}$ to denote variables in the uncorrelated space. To this end, we first decompose Σ as

$$\Sigma = VD^2V^T\,.$$

Here, V is an orthonormal matrix of eigenvectors of the covariance matrix Σ, and D is the diagonal matrix of the square root of the eigenvalues, or the ordered standard deviations in the basis spanned by the eigenvectors of Σ. We store the outputs from the pairs as columns of a matrix $Y_{kl} = (\mathcal{G}_T(\boldsymbol{\theta}_l))_k$, and then we change the basis of this matrix into the uncorrelated coordinates

$$\tilde{Y} = D^{-1}V^TY\,.$$

When trained on $\tilde{Y}$, GP returns the mean $\tilde{\mathcal{G}}_{\text{GP}}(\boldsymbol{\theta})$ and the (diagonal) covariance matrix $\tilde{\Sigma}_{\text{GP}}(\boldsymbol{\theta})$. We use tools from scikit-learn (Pedregosa et al., 2011) to train the emulator. After the diagonalization, we can train a scalar-valued GP for each of the 96 output dimensions, rather than having to train processes with vector-valued output. We construct a kernel by summing an Automatic Relevance Determination (ARD) radial basis function kernel and a white-noise kernel. The ARD kernel is a standard squared exponential kernel, where each input dimension has an independent lengthscale hyperparameter. This corresponds to regression, rather than interpolation, and the variance of the white noise kernel reflects the noise level assumed in the regression. We train by learning 4 hyperparameters: the radial basis function variance, a lengthscale for each of the two parameters $\boldsymbol{\theta}$ (due to ARD), and the white-noise variance. We train using the input–output pairs of the initial ensemble plus $N = 5$ subsequent iterations of the EKI algorithm. We use $M = 100$ ensemble members; thus, the training requires $(N+1)\times M = 600$ short (30-day) runs of our GCM.

We continue using the uncorrelated basis in the sampling stage; where required, we transform the output of the emulator back into a correlated basis,

$$\begin{aligned}\mathcal{G}_{\text{GP}}(\boldsymbol{\theta}) &= VD\tilde{\mathcal{G}}_{\text{GP}}(\boldsymbol{\theta}),\\ \Sigma_{\text{GP}}(\boldsymbol{\theta}) &= VD\tilde{\Sigma}_{\text{GP}}(\boldsymbol{\theta})DV^T\,.\end{aligned}$$

3.5 Sample: MCMC Sampling with a Gaussian Process Emulator

To quantify uncertainties, we use MCMC to sample the posterior distribution of parameters with the GP emulator. The primary reason for using the GP emulator goes back to the seminal paper by Sacks et al. (1989) and concerns the fact that it can be evaluated far more quickly than the GCM at a point in parameter space; this is important as we require more than 10^5 samples within the likelihood $\mathbb{P}(\boldsymbol{y} \mid \boldsymbol{\theta})$ in a typical MCMC run to sample the posterior distribution of parameters given data. However the emulator is also important for two additional reasons: (i) it naturally includes the approximation uncertainty (within $\tilde{\Sigma}_{\text{GP}}$) of using an emulator; (ii) it smooths the likelihood function because we work with an approximation of (6) based on the smooth $\mathcal{G}_\infty$, rather than (7) based on the noisy $\mathcal{G}_T$; as a result, MCMC is less likely to get stuck in local extrema.

Recall that we trained the GP in uncorrelated coordinates. Within MCMC, one can either map back into the original coordinates or continue working in the uncorrelated space. We choose to continue working in the uncorrelated space, and so we map each data realization $\boldsymbol{y}$ into this space: $\tilde{\boldsymbol{y}} = D^{-1}V^T\boldsymbol{y}$. In the Gaussian likelihood, we can use the GP emulated mean $\tilde{\mathcal{G}}_{\text{GP}}(\boldsymbol{\theta})$ and covariance matrix $\tilde{\Sigma}_{\text{GP}}(\boldsymbol{\theta})$ as surrogates for the map $\mathcal{G}_\infty$ and the internal variability covariance matrix Σ (after passing to the un-

correlated coordinates). That is, we approximate the Bayesian posterior distribution as

$$\begin{aligned}\mathbb{P}(\boldsymbol{\theta} \mid \tilde{\boldsymbol{y}}) &\propto \mathbb{P}(\tilde{\boldsymbol{y}} \mid \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta}) \\ &\propto \frac{1}{\sqrt{\det(\tilde{\Gamma}_{GP}(\boldsymbol{\theta}))}} \exp\left(-\frac{1}{2}\|\tilde{\boldsymbol{y}} - \tilde{\mathcal{G}}_{\mathrm{GP}}(\boldsymbol{\theta})\|^2_{\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta})}\right)\mathbb{P}(\boldsymbol{\theta}) \\ &\propto \exp\left(-\frac{1}{2}\|\tilde{\boldsymbol{y}} - \tilde{\mathcal{G}}_{\mathrm{GP}}(\boldsymbol{\theta})\|^2_{\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta})} - \frac{1}{2}\log\det\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta})\right)\mathbb{P}(\boldsymbol{\theta})\,. \end{aligned} \tag{8}$$

Here, $\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta}) = \tilde{\Sigma}_{\mathrm{GP}}(\boldsymbol{\theta}) + D^{-1}V^T\Delta V D^{-1}$ is the GP approximation of $\Gamma = \Sigma + \Delta$ in the uncorrelated coordinates. We include the (often overlooked) log-determinant term, arising from the normalization constant due to dependence of Γ_{GP} on $\boldsymbol{\theta}$. In the transformed parameter space, our prior $\mathbb{P}(\boldsymbol{\theta})$ is also Gaussian and therefore can be factored inside this exponential, adding a quadratic penalty to the negative log-likelihood. The MCMC objective function is then defined to be the negative logarithm of the posterior distribution (8); explicitly, given a Gaussian prior $N(\boldsymbol{m}, C)$ on the parameters, we define it as

$$\Phi_{\mathrm{MCMC}}(\boldsymbol{\theta}, \tilde{\boldsymbol{y}}) = \frac{1}{2}\|\tilde{\boldsymbol{y}} - \tilde{\mathcal{G}}_{\mathrm{GP}}(\boldsymbol{\theta})\|^2_{\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta})} + \frac{1}{2}\log\det\tilde{\Gamma}_{\mathrm{GP}}(\boldsymbol{\theta}) + \frac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{m}\|^2_C.$$

Evaluation of Φ_{MCMC} requires only the mean and covariance from the GP emulator. Furthermore Φ_{MCMC} is smooth and so is suitable for use within an MCMC algorithm to generate samples from the approximate posterior distribution of the parameters. Cleary et al. (2021) contains further discussion of MCMC using GPs to emulate the forward model, including situations where data comes from finite time-averages but the emulator is designed to approximate the infinite time-horizon forward model.

We use the random walk metropolis algorithm for MCMC sampling. The priors chosen were the same, physics-informed priors used to initialize EKI. We choose the proposal distribution also as a Gaussian with covariance proportional to the prior covariance. The MCMC run consists of a burn-in of 10,000 samples followed by 190,000 samples.

3.6 Benchmark Gaussian process (B-GP)

The performance of any emulator is dependent on the training points. Since we use an adaptive procedure (EKI) to concentrate the training points, which is the novel approach introduced in Cleary et al. (2021), we also train a benchmark emulator to compare our results with those resulting from more traditional, brute-force approaches to the emulation.

For this purpose, we use a GP emulator trained on a uniform set of points. Even in two dimensions, it is prohibitively costly for this set to span the support of the prior distribution. Instead, we use knowledge of the location and size of the posterior distribution to place a uniform grid of $40 \times 40 = 1600$ training points over $[-1.25, -0.5] \times [8.0, 10.0]$ in the transformed parameter space. This corresponds to $[0.62, 0.77] \times [0.83\text{ h}, 6.12\text{ h}]$ in the untransformed parameter space and captures the region of high probability of the posterior. The use of posterior knowledge here reduces the number of training points by a factor of 20 when compared to spanning 95% of the prior mass; no such posterior information is used in the CES algorithm, which automatically places training points where they are needed. The benchmark emulator uses the same kernel and training setup as in section 3.4, and we use the trained emulator in MCMC experiments in the same way as described in Section 3.5. To distinguish the two methods, we refer to the EKI-trained GP as EKI-GP and the benchmark (traditionally trained) GP as B-GP.

4 Results

To demonstrate the dependence of the parameter uncertainty on the realization of the (inflated) synthetic data, we reproduce the experiments 4 times with each of the four realizations shown in Figure 4. We denote these four sets of data $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$.

4.1 Calibrate: Ensemble Kalman Inversion

We use the first 6 iterations of EKI in the CES training process. The initial ensemble is spread over the whole parameter space but collapses within a few iterations near the true parameter values—to within 10% error in θ_{RH} and 30 minutes error in θ_τ (Figure 5). That is, the algorithm evolves toward consensus and toward the true solution. Biases arise from the realization of internal variability and the realization of the observational noise in each $\boldsymbol{y}^{(\cdot)}$.

To check for EKI convergence, we evaluate an additional 4 EKI iterations (labeled 0 to 9). At each iteration n, we compute residuals of the ensemble mean for each realization of the synthetic data $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$ created at the true parameters $\boldsymbol{\theta}^\dagger$,

$$\text{Residual}(n; \boldsymbol{y}^{(i)}) = \left\| \frac{1}{M} \sum_{m=1}^{M} \mathcal{G}_T(\boldsymbol{\theta}_m^{(n)}) - \boldsymbol{y}^{(i)} \right\|_\Gamma^2,$$

weighting the residuals by the covariance matrix Γ of the synthetic data. Figure 6(a) shows the residual as a function of EKI iteration. The residual decreases quickly over the first few iterations, before plateauing for subsequent iterations. Figure 6(b) shows standard deviations of the ensemble of parameters. The standard deviations decrease monotonically from iteration to iteration, reflecting the evolution toward consensus. The behavior is qualitatively similar for all realizations; quantitative differences reflect different realizations of internal variability in the different data realizations. This behavior reflects the fact that EKI is an optimization method for calibrating parameters: it is not constructed to learn uncertainty but rather to reach consensus around a single parameter value that makes the misfit small (Schillings & Stuart, 2017b).

4.2 Emulate: Validation

Figure 7 shows the parameter values used for training points for the EKI-GP and B-GP. We use the first 6 EKI iterations (i.e., 600 training points) for training. These are plotted over the associated objective function Φ_{MCMC}. The panels in the left column correspond to the EKI-GP using data vectors $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$. We see that the resulting objective functions Φ_{MCMC} from training EKI-GP at the marked training points (black dots) lead to unimodal distributions with a minimum near to a significant number of the training points; there are also training points that fall outside of the plotting domain (see Figure 5 for their extent). The right column of Figure 7 shows the benchmark grid for B-GP, which we use as a comparison for the EKI-GP method; the contours of Φ_{MCMC} were calculated using the same realization as their counterpart EKI-GPs. We see that for each realization the EKI-GP and B-GP produce objective functions that are qualitatively similar in terms of the magnitude of the minimum, the location of the minimum, and the approximate shape of the objective function; the quantitative differences are accounted for by differing geometry and density of training points (and hence a difference in approximation uncertainty). In both settings, the objective function Φ_{MCMC} is smooth because the GP smoothly approximates $\mathcal{G}_\infty$.

EKI-GP shows similar results for the objective function as B-GP, at a fraction of the computational effort. B-GP is far less practical as a methodology than is EKI-GP because it does not scale well to high-dimensional parameter spaces; it requires many more training points than EKI-GP. Even in our two-dimensional experiments, we needed to use posterior information to reduce the number of training points for B-GP by a fac-

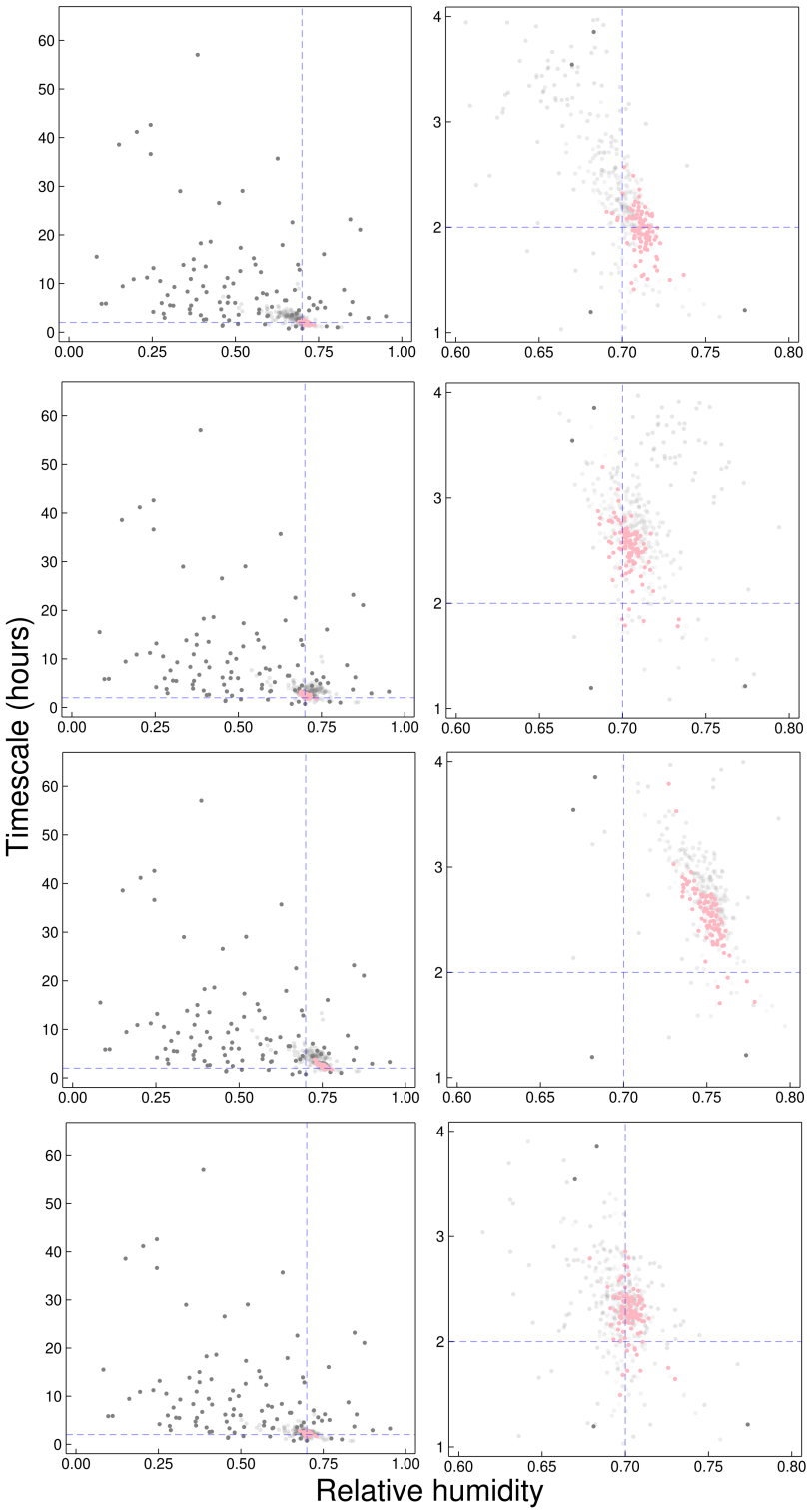

Figure 5. EKI ensemble at iterations 0 to 5 displayed as particles in parameter space. Left column: all members; right column: zoom-in near true parameter values. Each row represents optimization with a different data vector $\boldsymbol{y}^{(i)}$ from Figure 4. The (initial) prior ensemble 0 is highlighted in dark grey, and the final ensemble 5 is highlighted in pink. The intersection of the dashed blue lines represents the true parameter values used to generate observational data from the GCM.

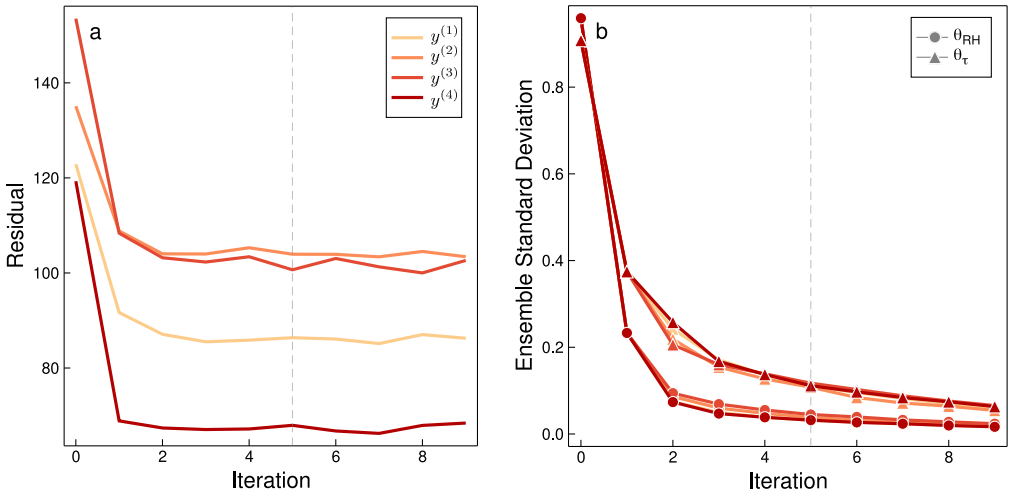

Figure 6. Convergence behaviour tests over 9 iterations of EKI for each realization of the data. The vertical dashed line marks the final iteration of Figure 5; we also show behaviour of 4 further iterations. (a) Ensemble-mean residuals relative to synthetic data for each EKI iteration. (b) Standard deviation of ensemble for the relative humidity parameter (circle) and timescale parameter (triangle) for each realization.

	σ_{RH}	σ_τ (hrs)
EKI (Iteration 9)	0.017	0.053
MCMC (EKI-GP)	0.099	0.265
MCMC (B-GP)	0.096	0.359

Table 1. Average standard deviations of parameters from EKI and MCMC experiments over $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$.

tor of 20. The B-GP comparison is included simply to demonstrate that EKI-GP achieves comparable results to those achieved by means of traditional emulation.

We validate the emulator approximation to the data by making a prediction at the true parameters $\boldsymbol{\theta}^\dagger$. We display $\mathcal{G}_{\text{GP}}(\boldsymbol{\theta}^\dagger)$ and the 95% confidence intervals computed using the variance from $\Sigma_{\text{GP}}(\boldsymbol{\theta}^\dagger)$ in Figure 8 for EKI-GP, and in Figure 9 for B-GP. The rows of Figure 8 correspond to the EKI-GP results for $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$. In both figures we also show the statistics of 600 30-day samples from the control simulation at $\boldsymbol{\theta}^\dagger$. Both the mean and 95% confidence intervals of all EKI-GP emulators (orange line and ribbon) closely match the statistics from the GCM runs (blue dots and error bars), as does the prediction from the B-GP (dark red line and ribbon). The training data are sufficient to ensure that the predicted 95% confidence interval from the emulators do not produce unphysical values (such as giving negative precipitation rates, or relative humidities outside $[0, 1]$).

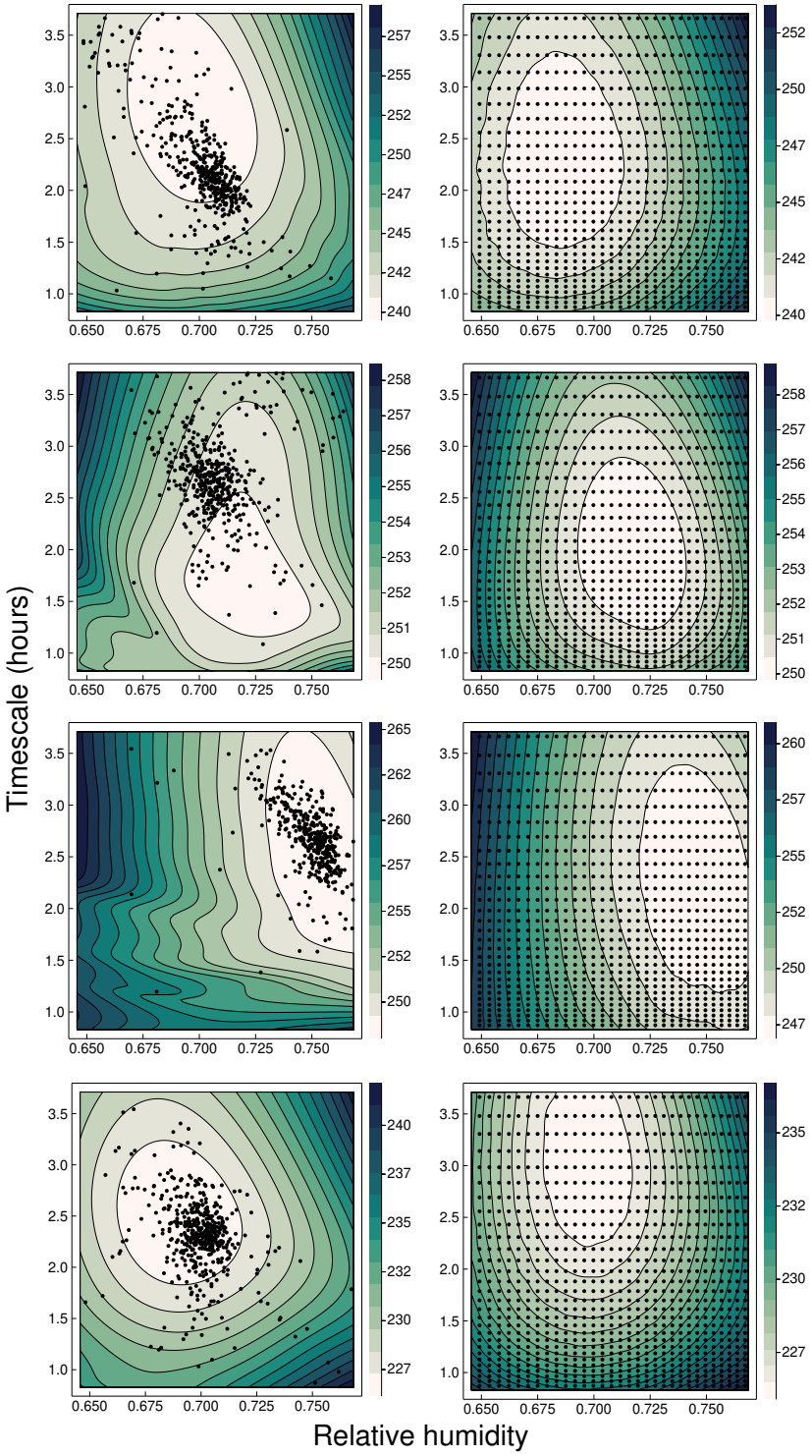

Figure 7. Training points for the GP emulators (EKI-GP and B-GP), plotted over the objective function Φ_{MCMC} for different data realizations $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$ (rows). Left column: particles representing members of the first 6 EKI iterations. Right column: grid (uniform in the transformed parameters) used to train the benchmark Gaussian process. In both cases, some additional training points fall outside of the plotting domain.

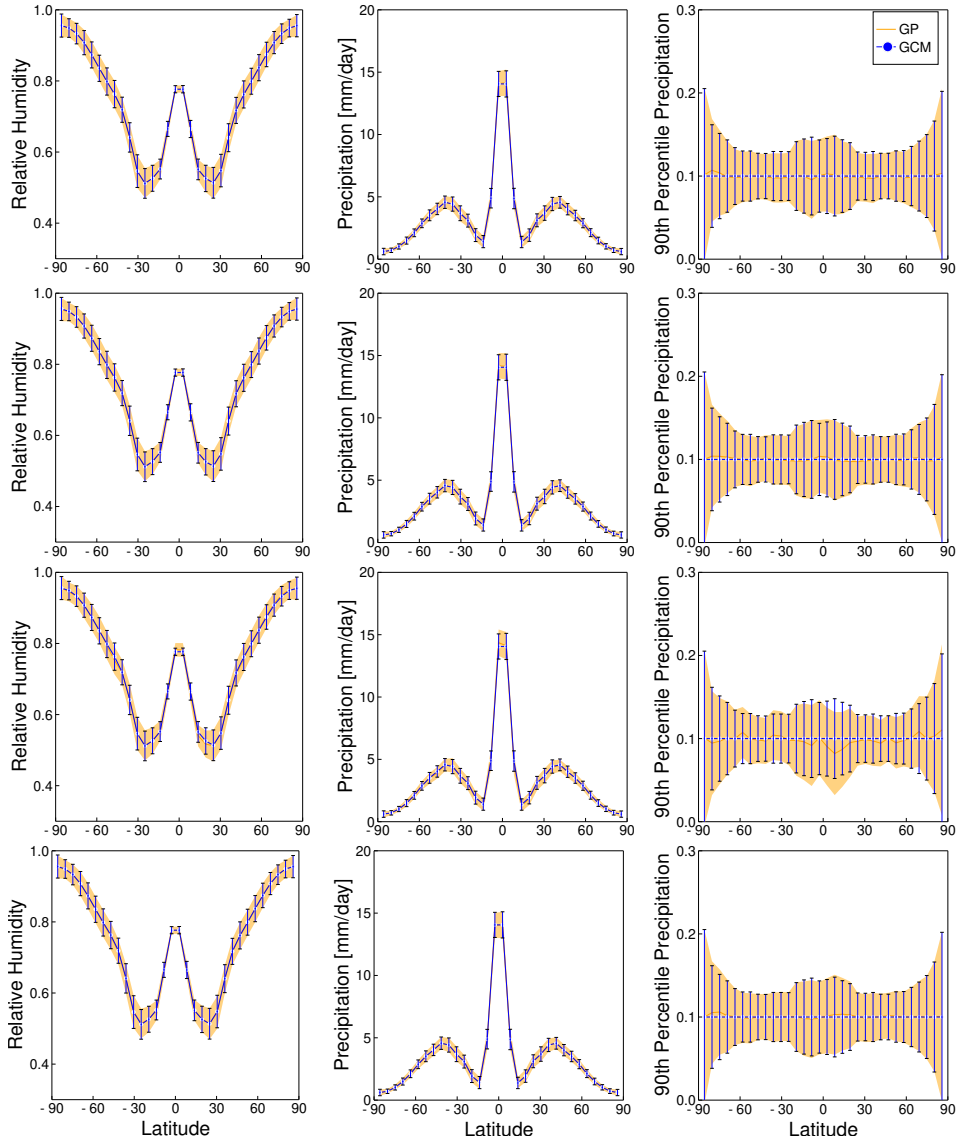

Figure 8. Comparison between the GCM statistics at the true parameters $\boldsymbol{\theta}^\dagger$ and the trained EKI-GP emulator at $\boldsymbol{\theta}^\dagger$. The four rows correspond to using EKI with the data vectors $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$. Blue lines: GCM mean (dots) averaged over 600 30-day runs, with the error bars marking a 95% confidence interval from variances on the diagonal of Γ. Orange: predicted mean (line) and 95% confidence interval (shaded region) produced by the GP emulator.

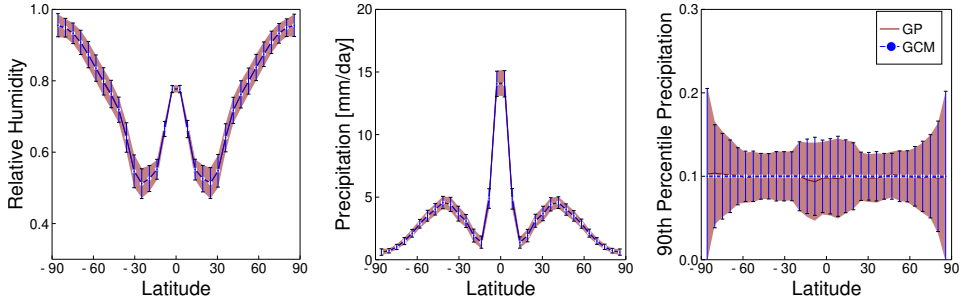

Figure 9. Comparison between the GCM statistics at the true parameters $\boldsymbol{\theta}^\dagger$ and the trained B-GP emulator predictions at $\boldsymbol{\theta}^\dagger$. Blue: GCM mean (dots) averaged over 600 30-day runs, with the error bars marking a 95% confidence interval from variances on the diagonal of Γ. Dark red: predicted mean (line) and 95% confidence interval (shaded region) produced by the B-GP emulator.

4.3 Sample: MCMC Sampling

We use an MCMC algorithm to generate a set of samples from the posterior distribution with the help of the GP emulator. We choose the random walk step size (which multiples the covariance in the proposal) at the start of a run to achieve proposal acceptance rates near to 25%. (This is near optimal in a precise sense for certain high-dimensional posteriors (Roberts et al., 2004); in practice, it works well beyond this setting.) All sampling is performed in the transformed space where the prior distribution is normal. Figure 10 shows kernel density estimates of the MCMC results; the panels in the left column are for EKI-GP (for $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$), and the panels in the right column are for B-GP for the same data realization. We display contours of the posterior that contain $50\%, 75\%$, and 99% of the mass of the posterior density.

All sets of results converge to similar regions of the parameter space about the true parameters, and the spread of uncertainty is quantified similarly in both EKI-GP and B-GP. Table 1 shows the standard deviations of the individual parameters alongside the empirical standard deviation calculated from the ensemble spread in EKI iteration 9. The standard deviations from the MCMC posterior based on EKI-GP and B-GP are similar. We re-emphasize that the EKI is constructed as an ensemble optimization method and has the property that the ensemble evolves towards consensus among the parameters, while also matching the data: the ensemble collapses. As a result, the EKI ensemble spread is an inadequate estimate the uncertainty, as seen in Table 1. As explained in the introduction, ensemble methods are only justifiable to quantify uncertainties in the Gaussian posterior setting (Chen & Oliver, 2012; Emerick & Reynolds, 2013). Our approach is justifiable whenever the GP accurately approximates the forward model (Cleary et al., 2021). The use of EKI for the design of training points for the GP does not require accurate uncertainty quantification within EKI; it only relies on EKI approximately locating minimizers of the model-data misfit.

There is sampling variability because of the different data realizations. This sampling variability can be assessed by asking which probability contours contain the true parameters. For both EKI-GP and B-GP, in three of four realizations, we capture the true values within 50% of the posterior probability mass; the realization $\boldsymbol{y}^{(3)}$ is captured only within the 99% contour of the posterior probability.

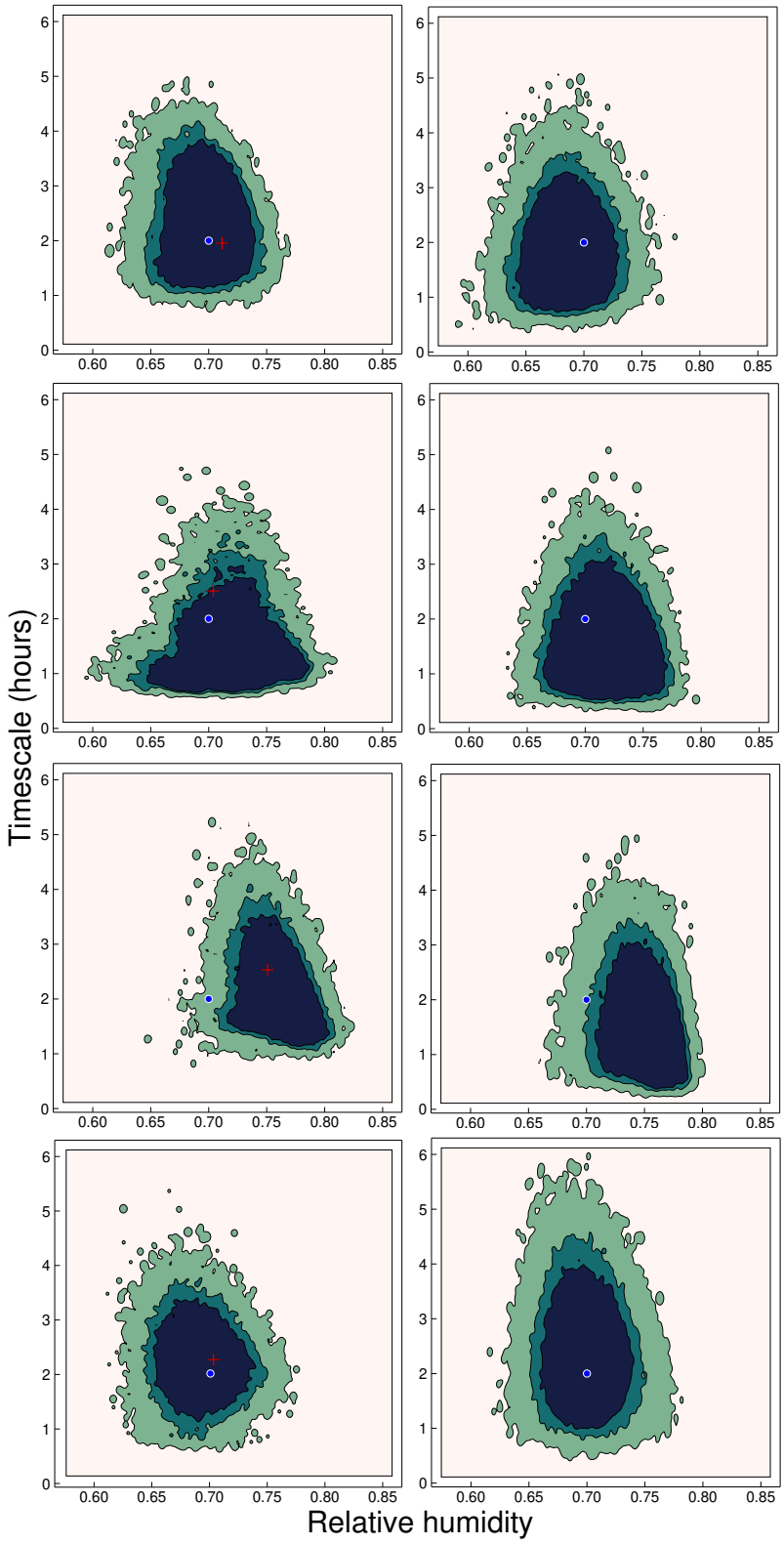

Figure 10. Density plot of MCMC samples of the posterior distribution. The contours are drawn to contain 50%, 75%, and 99% of the distribution generated from the samples. The left column show distributions learned using EKI-GP at $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(4)}$, and the right column using B-GP at the same data realizations. The blue dot represents the true parameters, while the red plus is the average across particles in the 6th EKI iteration.

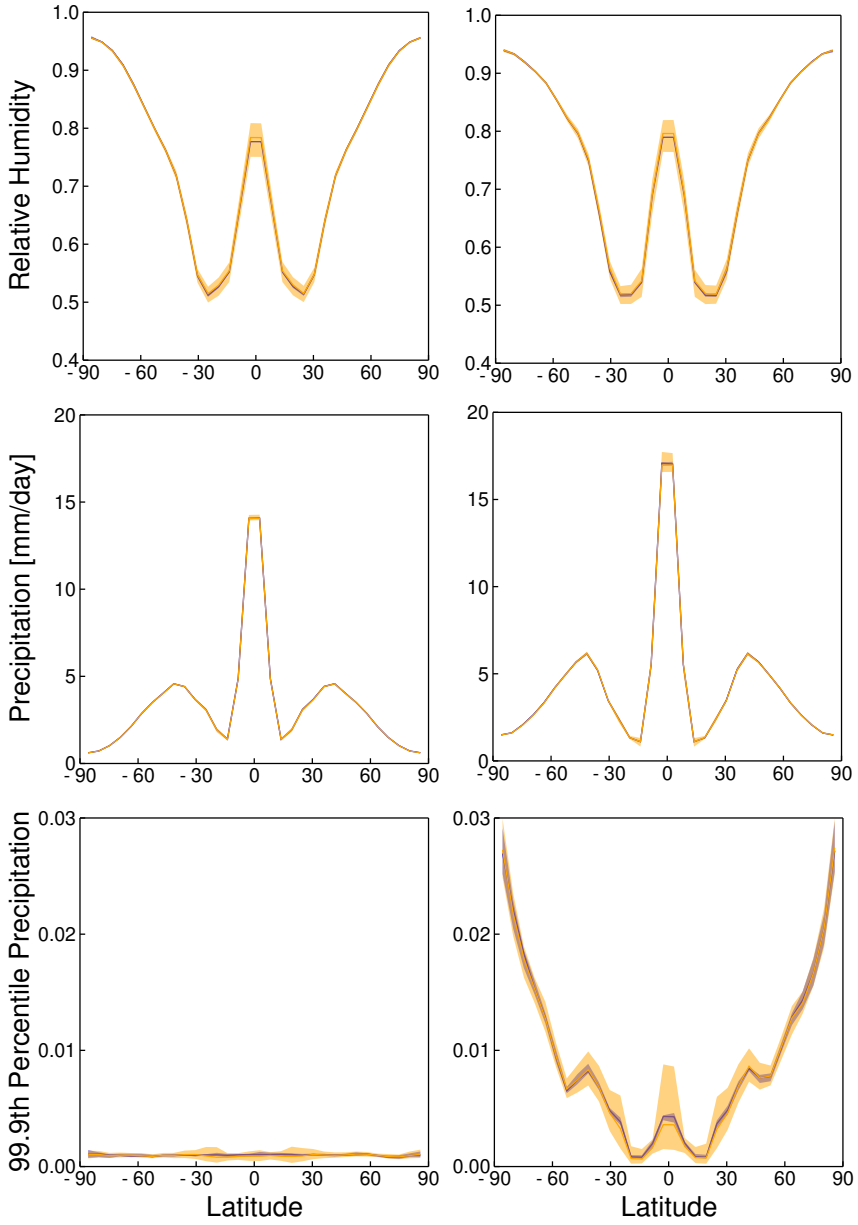

Figure 11. Comparison of statistics of a 7200-day average in a climate-change simulation. Left column: control climate; right column: warmer climate. Synthetic observational data evaluated at the true fixed parameters are shown in blue, while data evaluated at 100 samples from the posterior distribution (EKI-GP) are shown in orange. (We choose the posterior from the first data realization, top-left panel of Figure 10.) The solid lines are the medians, and the shaded regions represent the 95% confidence intervals (between the 2.5th and 97.5th percentiles). Top: Relative humidity in mid-troposphere. Middle: Precipitation rate. Bottom: Frequency with which 99.9th percentile of latitude-dependent daily precipitation in the control climate is exceeded.

4.4 Uncertainty Quantification in Prediction Experiments

To illustrate how the posterior distribution of parameters obtained in the sample step of the CES algorithm can be used to produce climate predictions with quantified uncertainties, we consider an idealized global-warming experiment. As in O'Gorman and Schneider (2008a, 2008b), we rescale the longwave opacity of the atmosphere everywhere by a uniform factor α. In the control climate we have considered so far, $\alpha = 1$. We generate a warm climate by setting $\alpha = 1.5$, which results in a global-mean surface air temperature increase from 287 K in the control climate to 294 K in the warm climate. To see parametric uncertainty rather than internal variability noise in the resulting "climate change predictions," we use long (7,200-day or approximately 20-year) averages in the prediction experiments.

We evaluate predictions of the latitude-dependent relative humidity and mean precipitation rate that we used in the CES algorithm. We also consider the frequency of precipitation extremes, now taken as the frequency with which the 99.9th percentile of daily precipitation in the control simulation is exceeded (rather than the 90th percentile we considered earlier). This last statistic indicates how the frequency of what are 1-in-1000 day precipitation events in the control climate change in the warmer climate.

We investigate the effect of parametric uncertainty on predictions by taking 100 samples of parameters from the posterior, creating a prediction for each sample, and comparing statistics of these runs with runs in which parameters are fixed to the true values $\boldsymbol{\theta}^\dagger$. The climate statistics in the control climate are shown in the left column of Figure 11. The runs from posterior samples (orange) and with fixed true parameters (blue) match well. The noise due to internal variability is quantitatively represented by the blue shaded region. Unlike in the earlier figures with short (30-day) averages (e.g., Figure 9), the internal variability noise here is small relative to the parametric uncertainty because of the (long) 7200-day averaging window. The orange shaded region contains both internal variability and parametric uncertainty and is dominated by parametric uncertainty. This remains the case in the warmer climate (right column of Figure 11).

The effects of global warming on atmospheric quantities is seen by comparing the two columns of Figure 11. Relative humidity is fairly robust to the warming climate, and precipitation rates increase globally (O'Gorman & Schneider, 2008b). The most dramatic changes occur for the frequency of extreme precipitation events (O'Gorman & Schneider, 2009b). What is a 1-in-1000 day event in the control climate (e.g., occuring with frequency 0.001) occurs in the extratropics of the warmer climate an order of magnitude more frequently, with the 95% confidence interval spanning 0.01 to 0.03. That is, a 1-in-1000 day event in the control climate occurs every 30 to 100 days in the warmer climate. The parametric uncertainty is particularly large for extreme precipitation events within the tropics—behavior one would not be able to see in global warming experiments with fixed parameters. This is consistent with the known high uncertainty in predictions of tropical rainfall extremes with comprehensive climate models (O'Gorman & Schneider, 2009a).

5 Conclusion and Discussion

The primary goal of this article was to demonstrate that ensemble Kalman inversion (EKI), machine learning, and MCMC algorithms can be judiciously combined within the calibrate-emulate-sample framework to efficiently estimate uncertainty of model parameters in computationally expensive climate models. We provided a proof-of-concept in a relatively simple idealized GCM.

Our approach is novel because we train a machine learning (GP) emulator using input-output pairs generated from an EKI algorithm. This methodology has several advantageous features:

1. It requires a modest number of runs of the expensive forward model (typically, $O(100)$ runs).
2. It generally finds optimal or nearly optimal parameters even in the presence of internal variability noise because EKI is robust with respect to such noise.
3. The resulting GP emulation is naturally most accurate around the (a priori unknown) optimal parameters because this is where EKI training points concentrate.
4. MCMC shows robust convergence to the posterior distribution, and allows identification of the optimal parameters with the maximum of the posterior probability, because it uses an objective function that is smoothed by GP emulation.

The effectiveness of GP depends on the training points, and a user must choose how many iterations of EKI to use for training (before ensemble collapse). In practice, we find the GP performance is robust as long as we include the initial iteration of training points (drawn from the prior) in our training set. The necessity of using the initial ensemble could be side-stepped by using an ensemble method that does not collapse, such as the recently introduced ensemble Kalman sampler (EKS) (Garbuno-Inigo et al., 2020).

The CES algorithm is efficient, as it addresses two dominant sources of computational expense. First, poor prior knowledge of model parameters requires blind exploration of a possibly high-dimensional parameter space to find optimal parameters and thus the region of high posterior probability. The CES framework handles this with an EKI algorithm, which we show to be successful when using time averaged data from a chaotic nonlinear model. Second, computing parametric uncertainty with a sampling technique (such as MCMC) generally requires many (10^5–10^6) evaluations of an expensive forward model. We instead solve a cheap approximate problem by exploiting GP emulators. We train the emulators on relatively few ($O(100)$) intelligently chosen evaluations provided by EKI, which ensures that training points are placed where they are most needed—near the minimum of the model-data misfit. The training itself introduces negligible computational cost relative to the running of the forward model, and the computational expense of evaluating the emulator in the sampling step is also negligible. Hence, the CES framework achieves about a factor 1000 speedup over brute-force MCMC algorithms. Significant efforts to accelerate brute-force MCMC without approximation have been undertaken (Järvinen et al., 2010; Solonen et al., 2012), and improvements of up to a factor 5 speedup have been made with adaptive and parallelized Markov chains. However, these approaches still are considerably more expensive than the CES algorithm.

The CES algorithm also has a smoothing property, which is beneficial even in situations where a forward model is cheap enough to apply a brute-force MCMC. If the forward model exhibits internal variability, the objective function for the sampling algorithm will contain a data misfit of the form (7), which has a random component because it contains a finite-time average. Without more sophisticated sampling methods, MCMC algorithms get stuck in noise-induced local minima. In the CES algorithm, only EKI uses the functional (7), and EKI is well suited for this purpose. The GP emulator learns the smooth, noiseless model $\mathcal{G}_\infty$ (in which internal variability disappears), using evaluations of $\mathcal{G}_T$ (which are affected by internal variability). Thus, MCMC within the CES algorithm uses the smooth GP approximation of (6).

The MCMC results in this study successfully capture the true parameters and their uncertainties. The results contain natural biases arising from the use of prior distributions, internal variability of the climate, and use of a single noisy sample as synthetic data. Despite the sampling variability and emulator constraints, our MCMC samples were able to capture the true parameters within an estimated 99% confidence interval in our examples, demonstrating the potential of EKI-trained GP emulators for MCMC sampling. Validation of the emulator (Figure 8) further supports the MCMC results, as do our comparisons with MCMC using the benchmark emulator (Table 1). The GP emulator both smooths the objective function and allows us to quantify uncertainty by sam-

pling from the posterior distribution. This contrasts with uncertainty quantification based on the EKI ensemble, which underestimates the true uncertainties in our experiments by an order of magnitude. As used here, EKI should be viewed as an optimization algorithm and not a sampling algorithm. Adding additional spread to match the posterior within EKI may be achieved for Gaussian posteriors (Chen & Oliver, 2012; Emerick & Reynolds, 2013) or by means of EKS (Garbuno-Inigo et al., 2020); however, these methods are not justifiable beyond the Gaussian setting. The MCMC algorithm within CES, on the other hand, samples from an approximate posterior distribution and is justifiable beyond the Gaussian posterior setting (Cleary et al., 2021).

Good scaling of the CES algorithm with the dimension of the parameter space will be of critical importance for moving beyond the current, low-dimensional proof-of-concept setting. Each stage of the CES algorithm scales to higher dimensions: For the calibration stage, ensemble methods scale well to high-dimensional state and parameter spaces, typically with $O(10^2)$ forward model runs (Kalnay, 2003; Oliver et al., 2008), if used with localization. In high dimensions, regularization is also typically needed in calibration algorithms, and various regularization schemes can be added (Chada et al., 2020; Iglesias, 2015, 2016; Garbuno-Inigo et al., 2019; Schneider et al., 2020b). The sampling stage also scales well to high dimensions. In particular, MCMC methods scale to non-parametric (infinite-dimensional) problems, in which the unknown is a function (Cotter et al., 2013). The current bottleneck for scalability lies with the GP emulator, which does not scale easily to high-dimensional inputs. However, other supervised machine learning techniques have potential to do so. For example, random feature maps and deep neural networks show promise in this regard (Nelsen & Stuart, 2020; Bhattacharya et al., 2020); incorporating these tools in the CES algorithm is a direction of current research.

An alternative form of constraining parameter uncertainty is history matching, or precalibration (Vernon et al., 2010; Edwards et al., 2011; Williamson et al., 2013). The idea complements that of Bayesian uncertainty quantification, where instead of searching for a high probability region of parameter space with respect to data, one rules out regions of parameter space that are deemed inconsistent with the data. Couvreux et al. (2020) and Hourdin et al. (2020) recently constrained the parameter space of a parameterization scheme by approximating a plausibility function over the parameter space using a Gaussian process, and then removing "implausible" regions of parameter space where the plausibility function passes a threshold. This removal process is iterated until the uncertainty of the emulator is small enough, or the space becomes empty. History matching accomplishes a similar adaptivity task as that performed in the CES algorithm by EKI. During early stages of history matching, however, one must sample the full parameter space with reasonable resolution, and emulator training is required at every iteration to evaluate the plausibility function. In contrast, in the CES algorithm, EKI draws a modest numbers of samples at every iteration and can work directly with noisy model evaluations, lowering the computational expense. The output of history matching is a (possibly empty) acceptable set of forward model runs; sampling this set leads to an upper bound on the prediction uncertainty. The benefit of the CES algorithm is that it provides samples of the posterior distribution, which lead to full estimates of prediction uncertainty (Figure 11). For this reason, history matching has been proposed as a preprocessing step for Bayesian uncertainty quantification, known as precalibration to improve priors and assess model validity (Vernon et al., 2010; Edwards et al., 2011). The CES algorithm targets the Bayesian posterior distributions directly.

In the more comprehensive climate modeling settings we target, data will be given from earth observations and from local high-resolution simulations (Schneider, Lan, et al., 2017). In these settings, model error leads to deficiencies when comparing model evaluations to data, leading to structural biases and uncertainty that must be quantified. Structural model errors can be quantified with a flexible hierarchical Gaussian process regression that learns a non-parametric form of the model deficiency, as demonstrated in pro-

totype problems in Schneider et al. (2020a). This approach represents model error in an interpretable fashion, as part of the model itself, rather than in the data space as pioneered in Kennedy and O'Hagan (2001).

The CES framework has potential for both the calibration and uncertainty quantification of comprehensive climate models, and other computationally expensive models. It is computationally efficient enough to use data averaged in time (e.g., over seasons), which need to be accumulated over longer model runs. Time-averaged climate statistics, including mean values and higher-order statistics such as extreme value statistics, are what typically matters in climate predictions. CES allows us to focus model calibration and uncertainty quantification on such immediately relevant statistics. Using time averaged statistics also has the advantage that it leads to smoother, albeit still noisy, objective functions when compared with calibration of climate models by minimizing mismatches in instantaneous, short-term forecasts (Schneider, Lan, et al., 2017). The latter approach can improve short-term forecasts but may not translate into improved climate simulations (Schirber et al., 2013). It also suffers from the difficulty that model resolution and data resolution may be mismatched. Focusing on climate statistics, as we did in our proof-of-concept here, circumvents this problem: time-aggregated climate statistics are varying relatively smoothly in space and, hence, minimizing mismatches in statistics between models and data does not suffer from the resolution-mismatch problem. CES can be used to learn about arbitrary parameters in climate models from time-averaged data. It leads to quantification of parametric uncertainties that then can be converted into parametric uncertainties in predictions by sampling from the posterior distribution of parameters.

Acknowledgments

This work was supported by the generosity of Eric and Wendy Schmidt by recommendation of the Schmidt Futures program, by the Hopewell Fund, the Paul G. Allen Family Foundation, and the National Science Foundation (NSF, award AGS1835860). A.M.S. was also supported by the Office of Naval Research (award N00014-17-1-2079). We thank Emmet Cleary for his preliminary work underlying some of the results shown here.

Data Availability. All computer code used in this paper is open source. The code for the idealized GCM, the Julia code for the CES algorithm, the plot tools, and the slurm/bash scripts to run both GCM and CES are available at https://doi.org/10.5281/zenodo.4393029.

References

Annan, J. D., & Hargreaves, J. C. (2007). Efficient estimation and ensemble generation in climate modelling. *Phil. Trans. R. Soc. A*, *365*, 2077–2088. doi: 10.1098/rsta.2007.2067

Betts, A. K. (1986). A new convective adjustment scheme. Part I: Observational and theoretical basis. *Quart. J. Roy. Meteor. Soc.*, *112*, 677–691.

Betts, A. K., & Miller, M. J. (1986). A new convective adjustment scheme. Part II: Single column tests using GATE wave, BOMEX, ATEX and arctic air-mass data sets. *Quart. J. Roy. Meteor. Soc.*, *112*, 693–709.

Betts, A. K., & Miller, M. J. (1993). The Betts–Miller scheme. In K. A. Emanuel & D. J. Raymond (Eds.), *The representation of cumulus convection in numerical models* (Vol. 24, pp. 107–121). Am. Meteor. Soc.

Bhattacharya, K., Hosseini, B., Kovachki, N. B., & Stuart, A. M. (2020). Model reduction and neural networks for parametric pdes. *arXiv preprint arXiv:2005.03180*.

Bischoff, T., & Schneider, T. (2014). Energetic constraints on the position of the Intertropical Convergence Zone. *J. Climate*, *27*, 4937–4951. doi: 10.1175/JCLI-D

-13-00650.1

Bocquet, M., & Sakov, P. (2012). Combining inflation-free and iterative ensemble kalman filters for strongly nonlinear systems. *Nonlinear Processes in Geophysics*, *19*(3), 383–399.

Bocquet, M., & Sakov, P. (2014). An iterative ensemble Kalman smoother. *Q. J. R. Meteorol. Soc.*, *140*, 1521–1535. doi: 10.1002/qj.2236

Bony, S., Colman, R., Kattsov, V. M., Allan, R. P., Bretherton, C. S., Dufresne, J.-L., ... Webb, M. J. (2006). How well do we understand and evaluate climate change feedback processes? *J. Climate*, *19*, 3445–3482. doi: 10.1175/JCLI3819.1

Bony, S., & Dufresne, J. L. (2005). Marine boundary layer clouds at the heart of tropical cloud feedback uncertainties in climate models. *Geophys. Res. Lett.*, *32*, L20806.

Bordoni, S., & Schneider, T. (2008). Monsoons as eddy-mediated regime transitions of the tropical overturning circulation. *Nature Geosci.*, *1*, 515–519. doi: 10.1038/ngeo248

Brient, F., & Schneider, T. (2016). Constraints on climate sensitivity from space-based measurements of low-cloud reflection. *J. Climate*, *29*, 5821–5835. doi: 10.1175/JCLI-D-15-0897.1

Cess, R. D., Potter, G., Blanchet, J., Boer, G., Ghan, S., Kiehl, J., ... others (1989). Interpretation of cloud-climate feedback as produced by 14 atmospheric general circulation models. *Science*, *245*, 513–516.

Cess, R. D., Potter, G. L., Blanchet, J. P., Boer, G. J., Del Genio, A. D., Déqué, M., ... Zhang, M.-H. (1990). Intercomparison and interpretation of climate feedback processes in 19 atmospheric general circulation models. *J. Geophys. Res.*, *95*, 16601–16615. doi: 10.1029/JD095iD10p16601

Chada, N. K., Stuart, A. M., & Tong, X. T. (2020). Tikhonov regularization within ensemble kalman inversion. *SIAM Journal on Numerical Analysis*, *58*(2), 1263–1294.

Chen, Y., & Oliver, D. S. (2012). Ensemble randomized maximum likelihood method as an iterative ensemble smoother. *Mathematical Geosciences*, *44*(1), 1–26.

Cleary, E., Garbuno-Inigo, A., Lan, S., Schneider, T., & Stuart, A. M. (2021). Calibrate, emulate, sample. *J. Comp. Phys.*, *424*, 109716. Retrieved from 10.1016/j.jcp.2020.109716

Cotter, S. L., Roberts, G. O., Stuart, A. M., & White, D. (2013). MCMC methods for functions: Modifying old algorithms to make them faster. *Statist. Science*, *28*(3), 424–446.

Couvreux, F., Hourdin, F., Williamson, D., Roehrig, R., Volodina, V., Villefranque, N., ... Xu, W. (2020). Process-based climate model development harnessing machine learning: I. A calibration tool for parameterization improvement. doi: 10.1002/essoar.10503597.1

Duncan, A. B., Stuart, A. M., & Wolfram, M.-T. (2021). Ensemble inference methods for models with noisy and expensive likelihoods. *arXiv preprint arXiv:2104.03384*.

Edwards, N. R., Cameron, D., & Rougier, J. (2011). Precalibrating an intermediate complexity climate model. *Climate Dynamics*, *37*(7-8), 1469–1482.

Emerick, A. A., & Reynolds, A. C. (2013). Ensemble smoother with multiple data assimilation. *Computers & Geosciences*, *55*, 3–15.

Ernst, O. G., Sprungk, B., & Starkloff, H.-J. (2015). Analysis of the ensemble and polynomial chaos Kalman filters in Bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, *3*(1), 823-851. doi: 10.1137/140981319

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, *99*(C5), 10143–10162.

Evensen, G. (2018). Analysis of iterative ensemble smoothers for solving inverse problems. *Comp. Geosci.*. doi: 10.1007/s10596-018-9731-y

Flato, G., Marotzke, J., Abiodun, B., Braconnot, P., Chou, S. C., Collins, W., ... Rummukainen, M. (2013). Evaluation of climate models. In T. F. Stocker et al. (Eds.), *Climate change 2013: The physical science basis. contribution of working group i to the fifth assessment report of the intergovernmental panel on climate change* (pp. 741–853). Cambridge, UK, and New York, NY, USA: Cambridge University Press.

Frierson, D. M. W. (2007). The dynamics of idealized convection schemes and their effect on the zonally averaged tropical circulation. *J. Atmos. Sci.*, *64*, 1959–1976.

Frierson, D. M. W., Held, I. M., & Zurita-Gotor, P. (2006). A gray-radiation aquaplanet moist GCM. Part I: Static stability and eddy scale. *J. Atmos. Sci.*, *63*, 2548–2566.

Garbuno-Inigo, A., Hoffmann, F., Li, W., & Stuart, A. M. (2019). Interacting Langevin Diffusions: Gradient Structure And Ensemble Kalman Sampler. , 1–34. Retrieved from `http://arxiv.org/abs/1903.08866`

Garbuno-Inigo, A., Hoffmann, F., Li, W., & Stuart, A. M. (2020). Interacting langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, *19*(1), 412-441. Retrieved from `https://doi.org/10.1137/19M1251655` doi: 10.1137/19M1251655

Geyer, C. J. (2011). Introduction to Markov Chain Monte Carlo. In S. Brooks, A. Gelman, G. L. Jones, & X.-L. Meng (Eds.), *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC Press. doi: 10.1201/b10905-3

Gland, F. L., Monbet, V., & Tran, V.-D. (2009). *Large sample asymptotics for the ensemble kalman filter* (Tech. Rep. No. RR-7014). INRIA.

Golaz, J.-C., Horowitz, L. W., & II, H. L. (2013). Cloud tuning in a coupled climate model: Impact on 20th century warming. *Geophys. Res. Lett.*, *40*, 2246–2251. doi: 10.1002/grl.50232

Gu, Y., & Oliver, D. S. (2007). An iterative ensemble kalman filter for multiphase fluid flow data assimilation. *SPE Journal*, *12*(04), 438–446.

Hourdin, F., Grandpeix, J.-Y., Rio, C., Bony, S., Jam, A., Cheruy, F., ... Roehrig, R. (2013). LMDZ5B: the atmospheric component of the IPSL climate model with revisited parameterizations for clouds and convection. *Clim. Dyn.*, *40*, 2193–2222. doi: 10.1007/s00382-012-1343-y

Hourdin, F., Mauritsen, T., Gettelman, A., Golaz, J.-C., Balaji, V., Duan, Q., ... Williamson, D. (2017). The art and science of climate model tuning. *Bull. Amer. Meteor. Soc.*, *98*, 589–602. doi: 10.1175/BAMS-D-15-00135.1

Hourdin, F., Williamson, D., Rio, C., Couvreux, F., Roehrig, R., Villefranque, N., ... Volodina, V. (2020). Process-based climate model development harnessing machine learning: II. Model calibration from single column to global. *Journal of Advances in Modeling Earth Systems*. doi: https://doi.org/10.1029/2020MS002225

Houtekamer, P. L., & Zhang, F. (2016). Review of the ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, *144*, 4489–4532. doi: 10.1175/MWR-D-15-0440.1

Iglesias, M. A. (2015). Iterative regularization for ensemble data assimilation in reservoir models. *Computational Geosciences*, *19*(1), 177–212.

Iglesias, M. A. (2016). A regularizing iterative ensemble kalman method for pde-constrained inverse problems. *Inverse Problems*, *32*(2), 025002.

Iglesias, M. A., Law, K. J. H., & Stuart, A. M. (2013, mar). Ensemble Kalman methods for inverse problems. *Inverse Problems*, *29*(4), 045001. doi: 10.1088/0266-5611/29/4/045001

Järvinen, H., Räisänen, P., Laine, M., Tamminen, J., Ilin, A., Oja, E., ... Haario, H. (2010). Estimation of ECHAM5 climate model closure parame-

ters with adaptive MCMC. *Atmos. Chem. Phys.*, *10*, 9993–10002. doi: 10.5194/acp-10-9993-2010

Kalnay, E. (2002). *Atmospheric modeling, data assimilation and predictability.* Cambridge University Press. doi: 10.1017/CBO9780511802270

Kalnay, E. (2003). *Atmospheric modeling, data assimilation and predictability.* Cambridge, UK: Cambridge Univ. Press.

Kaspi, Y., & Schneider, T. (2011). Winter cold of eastern continental boundaries induced by warm ocean waters. *Nature*, *471*, 621–624.

Kaspi, Y., & Schneider, T. (2013). The role of stationary eddies in shaping midlatitude storm tracks. *J. Atmos. Sci.*, *70*, 2596–2613.

Kennedy, M. C., & O'Hagan, A. (2001). Bayesian calibration of computer models. *J. Roy. Statist. Soc. B*, *63*, 425–464. doi: 10.1111/1467-9868.00294

Levine, X., & Schneider, T. (2015). Baroclinic eddies and the extent of the Hadley circulation: An idealized GCM study. *J. Atmos. Sci.*, *72*, 2744–2761. doi: 10.1175/JAS-D-14-0152.1

Li, G., & Reynolds, A. C. (2009). An iterative ensemble kalman filter for data assimilation. In (Vol. 14, pp. 496–505). Society of Petroleum Engineers.

Mauritsen, T., Stevens, B., Roeckner, E., Crueger, T., Esch, M., Giorgetta, M., ... Tomassini, L. (2012). Tuning the climate of a global model. *J. Adv. Model. Earth Sys.*, *4*, M00A01. doi: 10.1029/2012MS000154

Merlis, T. M., & Schneider, T. (2011). Changes in zonal surface temperature gradients and walker circulations in a wide range of climates. *J. Climate*, *24*, 4757–4768.

Nelsen, N. H., & Stuart, A. M. (2020). The random feature model for input-output maps between banach spaces. *arXiv preprint arXiv:2005.10224*.

O'Gorman, P. A. (2011). The effective static stability experienced by eddies in a moist atmosphere. *J. Atmos. Sci.*, *68*, 75–90.

O'Gorman, P. A., Lamquin, N., Schneider, T., & Singh, M. S. (2011). The relative humidity in an isentropic advection–condensation model: Limited poleward influence and properties of subtropical minima. *J. Atmos. Sci.*, *68*, 3079–3093.

O'Gorman, P. A., & Schneider, T. (2008a). Energy of midlatitude transient eddies in idealized simulations of changed climates. *J. Climate*, *21*, 5797–5806.

O'Gorman, P. A., & Schneider, T. (2008b). The hydrological cycle over a wide range of climates simulated with an idealized GCM. *J. Climate*, *21*, 3815–3832.

O'Gorman, P. A., & Schneider, T. (2009a). The physical basis for increases in precipitation extremes in simulations of 21st-century climate change. *Proc. Natl. Acad. Sci.*, *106*, 14773–14777.

O'Gorman, P. A., & Schneider, T. (2009b). Scaling of precipitation extremes over a wide range of climates simulated with an idealized GCM. *J. Climate*, *22*, 5676–5685.

Oliver, D. S., Reynolds, A. C., & Liu, N. (2008). *Inverse theory for petroleum reservoir characterization and history matching.* Cambridge Univ. Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Randall, D. A., & Wielicki, B. A. (1997). Measurements, models, and hypotheses in the atmospheric sciences. *Bull. Amer. Meteor. Soc.*, *78*, 400–406.

Reich, S. (2011). A dynamical systems framework for intermittent data assimilation. *BIT Numerical Mathematics*, *51*(1), 235–249.

Roberts, G. O., Rosenthal, J. S., et al. (2004). General state space Markov chains and mcmc algorithms. *Probability surveys*, *1*, 20–71.

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical science*, 409–423.

Sakov, P., Oliver, D. S., & Bertino, L. (2012). An iterative EnKF for strongly nonlinear systems. *Mon. Wea. Rev.*, *140*, 1988–2004. doi: 10.1175/MWR-D-11

-00176.1

Santner, T. J., Williams, B. J., Notz, W., & Williams, B. J. (2018). *The design and analysis of computer experiments* (2nd ed.). New York, NY: Springer.

Schillings, C., & Stuart, A. M. (2017a). Analysis of the ensemble kalman filter for inverse problems. *SIAM Journal on Numerical Analysis*, *55*(3), 1264-1290. Retrieved from `https://doi.org/10.1137/16M105959X` doi: 10.1137/16M105959X

Schillings, C., & Stuart, A. M. (2017b). Analysis of the ensemble Kalman filter for inverse problems. *SIAM J. Numer. Anal.*, *55*, 1264–1290.

Schirber, S., Klocke, D., Pincus, R., Quaas, J., & Anderson, J. L. (2013, mar). Parameter estimation using data assimilation in an atmospheric general circulation model: From a perfect toward the real world. *Journal of Advances in Modeling Earth Systems*, *5*(1), 58–70. doi: 10.1029/2012MS000167

Schmidt, G. A., Bader, D., Donner, L. J., Elsaesser, G. S., Golaz, J.-C., Hannay, C., ... Saha, S. (2017). Practice and philosophy of climate model tuning across six u.s. modeling centers. *Geosci. Model Dev..*, *10*, 3207–3223. doi: 10.5194/gmd-2017-30

Schneider, T., Lan, S., Stuart, A., & Teixeira, J. (2017). Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophys. Res. Lett.*, *44*, 12396–12417. doi: 10.1002/2017GL076101

Schneider, T., & O'Gorman, P. A. (2008). Moist convection and the thermal stratification of the extratropical troposphere. *J. Atmos. Sci.*, *65*, 3571–3583.

Schneider, T., O'Gorman, P. A., & Levine, X. J. (2010). Water vapor and the dynamics of climate changes. *Rev. Geophys.*, *48*, RG3001. (doi:10.1029/2009RG000302)

Schneider, T., Stuart, A. M., & Wu, J.-L. (2020a). Ensemble Kalman inversion for sparse learning of dynamical systems from time-averaged data. *arXiv preprint, arXiv:2007.06175*.

Schneider, T., Stuart, A. M., & Wu, J.-L. (2020b). Imposing sparsity within ensemble kalman inversion. *arXiv preprint arXiv:2007.06175*.

Schneider, T., Stuart, A. M., & Wu, J.-L. (2020c). Learning stochastic closures using ensemble Kalman inversion. *arXiv preprint, arXiv:2004.08376*.

Schneider, T., Teixeira, J., Bretherton, C. S., Brient, F., Pressel, K. G., Schär, C., & Siebesma, A. P. (2017). Climate goals and computing the future of clouds. *Nature Climate Change*, *7*, 3–5. doi: 10.1038/nclimate3190

Solonen, A., Ollinaho, P., Laine, M., Haario, H., Tamminen, J., & Järvinen, H. (2012). Efficient mcmc for climate model parameter estimation: Parallel adaptive chains and early rejection. *Bayesian Analysis*, *7*(3), 715–736. doi: 10.1214/12-BA724

Stephens, G. L. (2005). Cloud feedbacks in the climate system: A critical review. *J. Climate*, *18*, 237–273. doi: 10.1175/JCLI-3243.1

Van Leeuwen, P. J., & Evensen, G. (1996). Data assimilation and inverse methods in terms of a probabilistic formulation. *Monthly Weather Review*, *124*(12), 2898–2913.

Vernon, I., Goldstein, M., & Bower, R. G. (2010, 12). Galaxy formation: A Bayesian uncertainty analysis. *Bayesian Analysis*, *5*(4), 619–669. doi: 10.1214/10-BA524

Vial, J., Dufresne, J.-L., & Bony, S. (2013). On the interpretation of inter-model spread in CMIP5 climate sensitivity estimates. *Clim. Dyn.*, *41*, 3339–3362. doi: 10.1007/s00382-013-1725-9

Webb, M. J., Lambert, F. H., & Gregory, J. M. (2013). Origins of differences in climate sensitivity, forcing and feedback in climate models. *Clim. Dyn.*, *40*, 677–707. doi: 10.1007/s00382-012-1336-x

Wei, H.-H., & Bordoni, S. (2018). Energetic constraints on the ITCZ position in idealized simulations with a seasonal cycle. *J. Adv. Model. Earth Sys.*, *10*. doi: 10.1029/2018MS001313

Williamson, D., Goldstein, M., Allison, L., Blaker, A., Challenor, P., Jackson, L., & Yamazaki, K. (2013). History matching for exploring and reducing climate model parameter space using observations and a large perturbed physics ensemble. *Climate dynamics*, *41*(7-8), 1703–1729.

Wills, R. C., Levine, X. J., & Schneider, T. (2017). Local energetic constraints on Walker circulation strength. *J. Atmos. Sci.*, *74*, 1907–1922. doi: 10.1175/JAS-D-16-0219.1

Zhang, F., Sun, Y. Q., Magnusson, L., Buizza, R., Lin, S.-J., Chen, J.-H., & Emanuel, K. (2019). What is the predictability limit of midlatitude weather? *Journal of the Atmospheric Sciences*, *76*(4), 1077 - 1091. Retrieved from `https://journals.ametsoc.org/view/journals/atsc/76/4/jas-d-18-0269.1.xml` doi: 10.1175/JAS-D-18-0269.1

Zhao, M., Golaz, J.-C., Held, I. M., Guo, H., Balaji, V., Benson, R., ... Xiang, B. (2018). The GFDL global atmosphere and land model AM4.0/LM4.0: 2. Model description, sensitivity studies, and tuning strategies. *J. Adv. Model. Earth Sys.*, *10*, 735–769. doi: 10.1002/2017MS001209