



# Forward model emulator for atmospheric radiative transfer using Gaussian processes and cross validation

Otto Lamminpää<sup>1</sup>, Jouni Susiluoto<sup>1</sup>, Jonathan Hobbs<sup>1</sup>, James McDuffie<sup>1</sup>, Amy Braverman<sup>1</sup>, and Houman Owhadi<sup>2</sup>

<sup>1</sup>Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA

<sup>2</sup>Computing and Mathematical Sciences Department, California Institute of Technology, Pasadena, CA, USA

**Correspondence:** Otto Lamminpää (otto.m.lamminpaa@jpl.nasa.gov)

Received: 4 April 2024 – Discussion started: 3 May 2024

Revised: 19 November 2024 – Accepted: 28 November 2024 – Published: 6 February 2025

**Abstract.** Remote sensing of atmospheric carbon dioxide (CO<sub>2</sub>) carried out by NASA's Orbiting Carbon Observatory-2 (OCO-2) satellite mission and the related uncertainty quantification effort involve repeated evaluations of a state-of-the-art atmospheric physics model. The retrieval, or solving an inverse problem, requires substantial computational resources. In this work, we propose and implement a statistical emulator to speed up the computations in the OCO-2 physics model. Our approach is based on Gaussian process (GP) regression, leveraging recent research on kernel flows and cross validation to efficiently learn the kernel function in the GP. We demonstrate our method by replicating the behavior of OCO-2 forward model within measurement error precision and further show that in simulated cases, our method reproduces the CO<sub>2</sub> retrieval performance of OCO-2 setup with computational time that is orders of magnitude faster. The underlying emulation problem is challenging because it is high-dimensional. It is related to operator learning in the sense that the function to be approximated maps high-dimensional vectors to high-dimensional vectors. Our proposed approach is not only fast but also highly accurate (its relative error is less than 1 %). In contrast with artificial neural network (ANN)-based methods, it is interpretable, and its efficiency is based on learning a kernel in an engineered and expressive family of kernels.

## 1 Introduction

Climate change, one of the most significant global environmental challenges, is primarily attributed to anthropogenic carbon emissions, which have accelerated the increase of carbon dioxide (CO<sub>2</sub>) in the atmosphere, posing a threat to Earth's future. The industrial revolution marked the onset of increased CO<sub>2</sub> emissions due to the extensive use of fossil fuels in various industries, such as transportation, manufacturing, and agriculture. The Intergovernmental Panel on Climate Change underscores CO<sub>2</sub>'s potent effect on planetary warming due to significant radiative forcing (IPCC, 2023). The atmospheric concentration of this trace gas is increasing at an ever faster rate, and as of May 2023, the measured CO<sub>2</sub> at Mauna Loa station was 424.0 ppm, a 3 ppm increase from a year before (421.0 ppm in May 2022). Although the global terrestrial biosphere and oceans each take up about 25 % of these emissions (Friedlingstein et al., 2022), this balance may not be sustainable, which might lead to unpredictable feedbacks in the carbon cycle and the global climate system. These couplings between the Earth's climate system and the carbon cycle can introduce significant uncertainty in future climate change projections (Friedlingstein et al., 2014), which further renders mitigation efforts increasingly challenging.

For reliable climate modeling and future scenario prediction, it is crucial to estimate carbon flux accurately (e.g., CarbonTracker, Peters et al., 2007), which involves quantifying both the sources and natural sinks of carbon. However, current in situ measurement networks are primarily deployed in the northern midlatitudes, leaving areas like the tropics underrepresented (Schimel et al., 2015). This lack of extensive

coverage results in large uncertainties in flux estimates, underscoring the need for a more comprehensive global measurement network.

To provide a significant increase in coverage and resolution to the ground-based data set, global estimates of total column-averaged mole-fraction CO<sub>2</sub> (average amount of CO<sub>2</sub> over a vertical column of air at a specific ground pixel/location), denoted XCO<sub>2</sub>, are collected using satellite-borne spectrometers. These instruments include the Japanese Greenhouse gases Observing SATellite (GOSAT, Kuze et al., 2009), operational since January 2009; the follow-on GOSAT-2 (Imasu et al., 2023) launched in October 2018; the Orbiting Carbon Observatory-2 from NASA (OCO-2, Crisp et al., 2012), launched in July 2014; the OCO-3 instrument (Eldering et al., 2019) taken to the International Space Station in May 2019; and the Chinese TanSat (Ran and Li, 2019) and TanSat 2 (Wu et al., 2023). Planned future missions include the Geostationary Carbon Cycle Observatory (GeoCarb, Moore et al., 2018), the European CO<sub>2</sub> Monitoring Mission (CO2M, Sierk et al., 2019), and the Global Observing Satellite for Greenhouse gases and Water cycle (GOSAT-GW, Kasahara et al., 2020). In this work, we focus exclusively on OCO-2, which, like all the above-mentioned missions, measures solar radiance at the top of the atmosphere, reflected by Earth's surface and attenuated by atmospheric scattering and absorption by trace gases and aerosols. From these observed radiances, the OCO-2 mission uses a framework called *optimal estimation* (OE, Rodgers, 2004) to solve the related Bayesian inverse problem (see, e.g., Kaipio and Somersalo, 2005), referred to as a retrieval. OE is an iterative algorithm, returning an estimate of posterior mean and covariance as a Gaussian approximation to the nonlinear retrieval problem. Operationally, the retrieval problem is solved using the Atmospheric Carbon Observations from Space (ACOS) software (O'Dell et al., 2018), which implements OE using a state-of-the-art atmospheric full-physics (FP) model. Processing OCO-2 measurements with the ACOS algorithm is a computationally intensive task, and currently about one-third of prescreened clear soundings are used in a low-latency data processing stream (O'Dell et al., 2018). As the data record grows, computational speed is also a major hindrance for retrospective processing of the full collection of cloud-free soundings for the current and any future improved algorithms. Thus, computational efficiency is a limiting factor in releasing the improved data to the user community. These issues are certain to get even worse with upcoming wider-swath missions like CO2M and GOSAT-GW, as evidenced by another greenhouse gas imaging mission, Tropospheric Ozone Monitoring Instrument (TROPOMI, Veefkind et al., 2012), from regularly reprocessing their data record, which is more than 20 times greater in size than that of OCO-2.

As with all inverse problems, some approximations and assumptions have to be made in the ACOS algorithm. The resulting XCO<sub>2</sub> estimates have to be validated and bias-

corrected using ground-based measurements from the Total Carbon Column Observing Network (TCCON, Wunch et al., 2017) and the COllaborative Carbon Column Observing Network (COCCON, Frey et al., 2019) as a reference. These sites are concentrated on the northern midlatitudes, and as a result of this coverage issue and the imperfections in the FP model, significant systematic errors persist in the data set. (See, e.g., Kiel et al., 2019, for the effect of systematic errors and Cressie, 2018, for an overview of statistical treatment of and issues in the retrieval.) Considerable effort has been exerted to tackle the high accuracy (less than 0.3 parts per million (ppm) in scenes with background levels of around 410 ppm) and high precision (standard errors less than 0.5 ppm) requirements of ingesting OCO-2 into flux inversion, which is the primary application of the data product (Gurney et al., 2002; Patra et al., 2007; Liu et al., 2017; Palmer et al., 2019; Crowell et al., 2019; Peiro et al., 2022; Byrne et al., 2023). Recent advancements in applying Markov chain Monte Carlo (MCMC, Brynjarsdóttir et al., 2018; Lamminpää et al., 2019) for non-Gaussian posterior characterization and simulation-based uncertainty quantification (Braverman et al., 2021) for capturing the overall uncertainty in the retrieval pipeline have been successfully deployed for addressing persisting retrieval errors. These methods, although comprehensive, suffer equally from computational speed issues as they require an extensive number of FP evaluations.

Computational speed issues in OE retrievals have been addressed in several ways. Neural network (NN)-based machine learning approaches (David et al., 2021; Mishra and Molinaro, 2021; Bréon et al., 2022) have been implemented to a combination of real-world radiance data and model atmospheres (outputs of computational atmospheric models, like the Copernicus Atmospheric Monitoring Services (CAMS) model; Chevallier et al., 2010). The OCO-2 forward model itself was sped up by using a surrogate model (Hobbs et al., 2017) that only partially considered the physical processes present in the FP model and more recently by using a Gaussian process (GP) emulator (Ma et al., 2019) for replicating the output of the FP model. In this paper, we will take a similar approach using GPs but with several improvements and an application to solving the retrieval problem with the help of closed-form Jacobians required in the gradient-based algorithm. GPs are a well-suited technique for forward model emulation, since they can be used with less data (130 K, David et al., 2021, for NN vs. 20 K for GP; see Sect. 4) and trained more quickly than NN-based approaches. Additionally, a GP provides uncertainty estimates and closed-form Jacobians trivially, which are not straightforward to extract from a NN. For training efficiency, our approach will leverage recent novel techniques for GP parameter learning called *kernel flows* (Owhadi and Yoo, 2019) and training data generation via evaluating the FP model using the Reusable Framework for Atmospheric Composition (ReFRACtor) (McDuffie et al., 2020). We will demonstrate

the accuracy of forward model emulation against a holdout test set of FP evaluations and further demonstrate the ability of our emulator to replicate the OE retrieval performance of ReFRACtor FP model in a fraction of the computational time. Our approach achieves a remarkably low prediction error, less than 1 % (“within measurement error limits”), which is an excellent result in the field of more general *operator learning*. Strategies to achieve learning more complicated operators, like the FP model in our case, often involve a NN-based architecture (Lu et al., 2021; Li et al., 2024). Our approach follows the example set by Batlle et al. (2023) that kernel methods are competitive in operator learning.

The rest of the paper is organized as follows. Section 2 will describe in detail the GP regression, kernel learning, and the resulting forward model emulator. Section 3 will further elaborate on the details of the OCO-2 retrieval algorithm, the state vector, and the FP model describing atmospheric radiative transfer. Section 4 will detail the emulator implementation of the ReFRACtor FP model and assess its performance. Section 5 will show results of our emulator used in a simulated XCO<sub>2</sub> retrieval context, and finally Sect. 6 will provide concluding remarks and ideas on future work and applications.

## 2 Gaussian process emulator

*Gaussian process (GP) regression* (Rasmussen and Williams, 2006) (also called *kriging* in spatial context: Cressie, 1993; Stein, 1999) is a well-studied methodology for approximating any continuous function to an arbitrary accuracy, leveraging training data and a *kernel function* prescribed a priori. In addition, once trained, the GP model can be used to obtain fast and accurate predictions of a computationally demanding physics model, to estimate prediction uncertainty, and to compute closed-form derivatives and Jacobians for the prediction. Physical constraints like positive parameter values can be accounted for in training data design (which we address in later sections) so that predictions happen with the support of a training data set. This is to say that our training data set will cover the expected minimum and maximum values of each parameter in the state vector. For example, surface albedo is physically restricted between 0 (no reflected light) and 1 (full reflection). If our training data are sufficiently well spread covering this range, we can make good predictions essentially by interpolation with the physically feasible interval. Potential departure from this support can be detected by large prediction uncertainty values, as the prediction uncertainty of a GP gets large by design if the input location does not have training points near it. In this section, we outline the basic theory of GP regression and outline our approach to modeling the continuous function between atmospheric state vectors  $\mathbf{x}$  and radiances  $\mathbf{y}$  observed by the OCO-2 instrument. We also provide background on maximum likelihood estimation

for fitting GP models and present a novel root mean square error (RMSE) cross-validation extension for the kernel flow (Owhadi and Yoo, 2019) approach, which we employ for the rest of this work.

### 2.1 Gaussian process regression

To construct an emulator for the forward model  $F(\mathbf{x})$ , we employ Gaussian process (GP) regression to predict a *label*  $z^* \in \mathbb{R}$  at a new *state*  $\mathbf{x}^* \in \mathbb{R}^m$ . A GP is defined by a *kernel function* (defined explicitly later)  $k(x, x') : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ , where in the cases studied in this work  $\mathcal{X} = \mathbb{R}^m$ . We denote by  $\Gamma[\mathbf{X}, \mathbf{X}]$  the matrix of all kernel function evaluations over the training data  $\mathbf{X} \in \mathbb{R}^{m \times N}$  of  $N$  points with the entries  $\Gamma[\mathbf{X}, \mathbf{X}]_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$ , where  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are the  $i$ th and  $j$ th training data points, respectively. Furthermore,  $\Gamma[\mathbf{x}^*, \mathbf{X}]$  denotes the vector of kernel evaluations of state  $\mathbf{x}^*$  against all training points  $\mathbf{X}$ . Using the training data together with vector of corresponding labels  $\mathbf{z} \in \mathbb{R}^N$ , a GP prediction of label (or function value) at a new state  $\mathbf{x}^*$  is given by

$$z^* \equiv \text{GP}(\mathbf{x}^*) = \Gamma[\mathbf{x}^*, \mathbf{X}](\Gamma[\mathbf{X}, \mathbf{X}] + \sigma \mathbb{I})^{-1} \mathbf{z}, \quad (1)$$

where we have assumed without loss of generality that the training data are centered, and thus the GP has a zero mean. We add that the term  $(\Gamma[\mathbf{X}, \mathbf{X}] + \sigma \mathbb{I})^{-1} \mathbf{z}$  does not depend on the new input state, so it can be precomputed. This makes the predictions take minimal computational time by avoiding inverting a potentially large matrix  $(\Gamma[\mathbf{X}, \mathbf{X}] + \sigma \mathbb{I})^{-1}$ .

In GP literature, the variance term  $\sigma \mathbb{I}$  is usually taken to be the measurement error or local-scale unexplained variability in the training labels  $\mathbf{z}$ . However, since we are interested in reproducing the outputs of a computer code, the “measurements” are exact, and hence there is no measurement error. It was shown in Owhadi and Yoo (2019) that learning the parameters of GP models from noiseless data can lead to unstable predictive models and numerical singularities. For this reason, we treat  $\sigma$  as a regularization parameter, which captures the empirical mismatch between the model and the actual data, and optimize it together with other kernel parameters.

In addition to point predictions, GP prediction can be associated with prediction uncertainty (the posterior variance of the GP), given by

$$\sigma^* = k(\mathbf{x}^*, \mathbf{x}^*) - \Gamma[\mathbf{x}^*, \mathbf{X}](\Gamma[\mathbf{X}, \mathbf{X}] + \sigma \mathbb{I})^{-1} \Gamma[\mathbf{x}^*, \mathbf{X}]^T. \quad (2)$$

The ability to include prediction uncertainties sets GP regression apart from many modern NN-based machine learning methods, which only provide a point estimate as a prediction. Large prediction variance can be an indication of departure from the support of a training data set, indicating that GP is likely to lose its prediction skill. Additionally, uncertainty from the predictions can be propagated forward and accounted for in further applications of GP-based emulators.

The GP formulas presented here rely on conditional Gaussian distributions and thus have a similar structure to that of

optimal interpolation (OI; e.g., p. 157 of Kalnay, 2002) and related iterative optimal estimation (OE) algorithms (e.g., Eq. 15), both of which are widely used methods in atmospheric remote sensing. OI and OE use Gaussian assumptions to derive a mean and covariance for the posterior distribution as a solution to an inverse problem, i.e., data assimilation or a retrieval. In Gaussian process regression, the target function (here, the forward model) is represented similarly by a Gaussian distribution that has a mean (prediction) and covariance (error estimate of the prediction).

Our interest will be in replicating the results of a gradient-based optimization problem. Hence, in addition to fast evaluations of  $F(\mathbf{x})$ , we would also benefit from fast derivatives obtained from closed-form expressions. Combining Eqs. (1) and (4), we get

$$\frac{d}{d\mathbf{x}^*} z^* = \frac{d}{d\mathbf{x}^*} \Gamma[\mathbf{x}^*, \mathbf{X}] (\Gamma[\mathbf{X}, \mathbf{X}] + \sigma \mathbb{I})^{-1} \mathbf{z}, \quad (3)$$

which describes taking the derivative of Eq. (1) with respect to  $\mathbf{x}^*$ .

While other machine learning methods, such as artificial neural networks and multilayer perceptrons, can in principle be differentiated, computing the derivatives of a large architecture is computationally more demanding than evaluating Eq. (3), which motivates our use of GP regression. Other similar approaches (e.g., radial basis function networks) can be shown to be universal approximators as well and could be used in place of GPs. As will be shown, our approach yields a fast and accurate predictor that is intuitive and relatively easy to implement, so comparison against other machine learning methods will not be pursued further in the scope of this work.

## 2.2 Kernel function

A crucial modeling choice in GP regression is specification of a kernel function. This task involves either expert knowledge of the domain structure or some iterative trial-and-error search. In our application, we have empirically observed that a kernel function consisting of the sum of Matérn and linear kernels yields excellent predictive performance. This is likely due to a locally near-linear behavior commonly assumed with the OCO-2 forward model being captured by the linear kernel, together with a largely flexible Matérn term that is known to capture a large variety of nonlinear effects. The Matérn kernel is a more expressive choice of kernel compared to the usual Gaussian/radial basis functions used by default in Gaussian process regression, which tend to be “too smooth” to capture more abrupt changes in the function that is being approximated. Furthermore, such a kernel can also be differentiated in closed form. The kernel function used

throughout this work is given by

$$k(\mathbf{x}, \mathbf{x}') = \alpha_1 \left( 1 + \frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \exp \left( -\frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) + \alpha_2 (\mathcal{W}\mathbf{x})^T (\mathcal{W}\mathbf{x}'), \quad (4)$$

where  $\|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} = \sqrt{(\mathbf{x} - \mathbf{x}')^T \mathcal{W}^2 (\mathbf{x} - \mathbf{x}')}$ ,  $\mathcal{W} = \text{diag}(\mathbf{w})$  is a diagonal matrix,  $\mathbf{w} \in \mathbb{R}^m$  is a vector of weights,  $l \in \mathbb{R}$  is a length scale parameter, and  $\alpha_1$  and  $\alpha_2 \in \mathbb{R}_+$  are positive weights that are restricted to sum to 1.

To compute Jacobians, we need an expression for the derivative of the kernel function  $\frac{d}{d\mathbf{x}^*} \Gamma[\mathbf{x}^*, \mathbf{X}]$  in Eq. (3). This can be computed in closed form from Eq. (4) using known matrix identities. The derivation of a closed-form expression can be found in Appendix A.

## 2.3 Parameter learning

Prediction quality of GP regression depends on identifying the hyperparameters  $\theta$  that best fit the training data. In our case, following the form of our kernel function, we have  $\theta = [\mathbf{w}, l, \sigma, \alpha_1, \alpha_2]$ . Hyperparameters are commonly learned via optimization, using maximum likelihood estimation (MLE, Rasmussen and Williams, 2006). This amounts to minimizing

$$\mathcal{L}(\theta) = -\frac{1}{2} \log [\det(\Gamma_\theta)] - \frac{1}{2} \mathbf{z}^T \Gamma(\theta)^{-1} \mathbf{z}, \quad (5)$$

where  $\Gamma(\theta) = \Gamma[\mathbf{X}, \mathbf{X}]$  evaluated at parameter values  $\theta$ . Although this method is usually robust and performs well, GP applications with high-dimensional inputs and a large amount of training data are known to be challenging due to inverse matrix and log-determinant calculations. Numerous approaches have been suggested to tackle this problem (e.g., local approximations, Vecchia, 1988; Datta et al., 2016). Inspired by the kernel flow approach (Owhadi and Yoo, 2019) where kernel parameters are learned by minimizing a relative reproducing kernel Hilbert space (RKHS) norm, we propose a cross-validation RMSE-based method to be used in this work. Intuitively, RKHS norm is a way to measure the smoothness of a function approximation achieved with the kernel method. While smooth methods generally yield discretization-invariant predictors, we propose to directly minimize the prediction error instead. The intuition of this approach is to iteratively select small mini-batches of the training data set and individually leave points out one by one while using the rest of the mini-batch to predict the left-out values (via Eq. 1). Learning the kernel parameters that minimize this prediction error leads to globally good predictions given new inputs. This approach leverages the known screening effect associated with Matérn kernels, where the effects of faraway points on prediction accuracy diminish, and only close-by points are necessary for prediction accuracy. The

same intuition is the basis of numerous nearest neighbors GP methods (e.g., Vecchia, 1988). The upside of our approach is the ability to select small mini-batches on each training iteration, allowing for faster computations while avoiding expensive log-determinant calculations and inverting the large covariance matrices required in MLE. We will later show that our proposed method converges reliably and yields excellent predictions.

We start by selecting a mini-batch  $\mathbf{X}^{\text{batch}}$  and  $\mathbf{z}^{\text{batch}}$  of size  $N_{\text{batch}}$  by randomly sampling from training data. We define a leave-one-out (LOO) cross-validation loss function with respect to  $L^2$  error (also known as RMSE) by first considering taking out one data point from the training data and using the rest to predict it. This can be achieved by modifying the GP prediction formula from Eq. (1) and leaving out the  $i$ th data point. This is achieved via a rank-one downdate  $\tilde{\Gamma}(\theta)^{-1} - \frac{\tilde{\Gamma}(\theta)^{-1}_{:,i} \tilde{\Gamma}(\theta)^{-T}_{:,i}}{\tilde{\Gamma}(\theta)^{-1}_{i,i}}$  to remove the effect of the  $i$ th data point from the inverse covariance matrix  $\tilde{\Gamma}(\theta)^{-1}$ . (See Stewart, 1998, and Zhu et al., 2022, for details.) The modified LOO prediction formula is then given by

$$\tilde{\text{GP}}(\theta, i) = \tilde{\Gamma}(\theta)^T_{:,i} \left( \tilde{\Gamma}(\theta)^{-1} - \frac{\tilde{\Gamma}(\theta)^{-1}_{:,i} \tilde{\Gamma}(\theta)^{-T}_{:,i}}{\tilde{\Gamma}(\theta)^{-1}_{i,i}} \right) \mathbf{z}^{\text{batch}}, \quad (6)$$

where  $\tilde{\Gamma}(\theta) = \Gamma(\mathbf{X}^{\text{batch}}, \mathbf{X}^{\text{batch}})$  is the  $N_{\text{batch}} \times N_{\text{batch}}$  covariance over the mini-batch evaluated at parameter values  $\theta$ . Here, the notation  $\tilde{\Gamma}(\theta)_{:,i}$  means all rows of the  $i$ th column. We then define the final loss function by using Eq. (6) to predict  $z_i$  (the  $i$ th training label removed from the mini-batch) as

$$\rho(\theta) = \sum_{i=k_1}^{k_p} (\tilde{\text{GP}}(\theta, i) - z_i)^2 + \epsilon \|\theta_0 - \theta\|^k, \quad (7)$$

where  $i \in [k_1 \dots k_p] \subset [1 \dots N_{\text{batch}}]$  is a subset of  $p \leq N_{\text{batch}}$  indices denoting elements of the mini-batch selected for prediction, which can be chosen as, for example, the entire mini-batch or the  $p$  nearest neighbors of the center point of the mini-batch. The regularization term, with error norm  $\|\cdot\|^2$ , some penalty magnitude  $\epsilon$ , and mean  $\theta_0$ , is included to ensure that kernel amplitude parameter values do not grow uncontrollably. This is done since we have observed empirically that letting non-identifiable parameters grow during optimization can lead to the optimizer getting “stuck”, whereas this problem is not observed when regularizing the loss function. One may, for example, set  $\theta_0$  to be a vector of 1’s.

We can now optimize the kernel parameters iteratively by repeatedly selecting mini-batches and updating  $\theta$  along the gradient of  $\rho(\theta)$ , which is obtained by automatic differentiation using Julia’s *Zygote* package (Innes, 2019). We note that closed-form kernel derivatives could be used here as well, but since automatic differentiation with mini-batch sizes we use uses negligible computational time, we will not pursue this idea further in this work. We note that as the mini-batch is

selected at random, this method can be viewed as stochastic gradient descent. For this reason, we use the adaptive moment estimation (ADAM, Kingma and Ba, 2017) optimizer to find the optimal value. Use of a momentum-based optimizer is further recommended in this application as we have observed that the cost function often has several local minima. The optimization procedure is summarized in Algorithm 1. The final parameter value can be selected to be the one corresponding to the smallest loss function value achieved during training.

---

**Algorithm 1** Kernel parameter learning

---

**Input:** kernel function  $k$ , training data  $(\mathbf{X}, \mathbf{z})$ , batch size  $N_{\text{batch}}$ , number of prediction points  $p$ , number of iterations  $N_{\text{Iter}}$ .  
**Output:** matrix of kernel parameters  $\Theta$  and vector of loss values  $\mathbf{R}$

- 1: initialize  $\theta_1 \leftarrow 1, \Theta \leftarrow \mathbf{0}, \mathbf{R} \leftarrow \mathbf{0}$
- 2: **for all**  $i$  in  $1 \dots N_{\text{Iter}}$  **do**
- 3:  $\mathbf{X}^{\text{batch}} \leftarrow \mathbf{X}[\text{rand}(N_{\text{batch}})], \mathbf{z}^{\text{batch}} \leftarrow \mathbf{z}[\text{rand}(N_{\text{batch}})]$  // Randomly select a mini-batch  $\mathbf{X}^{\text{batch}}, \mathbf{z}^{\text{batch}}$
- 4:  $\mathbf{R}[i] \leftarrow \rho(\theta_i)$  // Compute loss  $\rho(\theta_i)$  from Eq. (7)
- 5:  $\Theta[i] \leftarrow \theta_{i+1}, \theta_{i+1} \leftarrow \theta_i + \text{ADAM}(\frac{\partial}{\partial \theta} \rho(\theta_i))$  // Compute gradient  $\frac{\partial}{\partial \theta} \rho(\theta_i)$  and update parameters  $\theta_i$  using ADAM
- 6: **end for**
- 7: **return**  $\Theta, \mathbf{R}$

---

**2.4 Training data generation**

As we aim to reproduce the performance of a function represented as computer code, we take advantage of the freedom to use a space-filling design for  $\mathbf{x}$  in  $\mathbb{R}^m$  for training data creation. We first span the unit cube  $[0, 1]^m$  with a *Sobol’ sequence* (Sobol, 1967; Press et al., 1992) of  $N$  points. In practice we employ Julia’s *Sobol.jl* (Johnson, 2020) package for this step. Then, using information about the minimum and maximum physically feasible value of each input dimension, we scale the unit cube to span the whole state space. During research, we tested other methods like random sampling and Latin-hypercube-based methods, which turned out to leave “holes” in training data set, meaning non-constant predictive performance over the entire data set. Sobol’ sequences, meanwhile, span the entire input space more evenly. Sobol’ sequences are a space-filling design akin to Latin hypercubes, providing optimality (observed experimentally) in generation of training data. We further evaluate the computational model  $F(x)$  at each training point, obtaining states  $\mathbf{X} \in \mathbb{R}^{N \times m}$  and model outputs  $\mathbf{Y} \in \mathbb{R}^{N \times n}$ .

**3 The Orbiting Carbon Observatory-2**

In this section, we describe OCO-2 and the related measurements, physics model, state vector, and retrieval algorithm. Further information on these topics can be found in, for example, Connor et al. (2008), O’Dell et al. (2012), Crisp et al.

(2012), O'Dell et al. (2018), and in the Algorithm Theoretical Basis Document (ATBD) (Boesch et al., 2015).

### 3.1 The OCO-2 instrument

OCO-2 is a NASA-operated satellite mission dedicated to providing data products of global atmospheric carbon dioxide concentrations (Crisp et al., 2004). The satellite is pointed towards Earth as it measures solar light reflected by Earth's surface and atmosphere, recorded as radiances. The OCO-2 instrument itself is composed of three spectrometers that measure light reflected from Earth's surface in the infrared part of the spectrum in three separate wavelength bands. These bands are centered around 0.765, 1.61, and 2.06  $\mu\text{m}$  and are called the O<sub>2</sub> A-band (O2), the weak CO<sub>2</sub> band (WCO2), and the strong CO<sub>2</sub> band (SCO2), respectively. Each observation consists of 1016 radiances on separate wavelengths from each band (for more information, see, e.g., Crisp et al., 2017; Rosenberg et al., 2017). These measurements are then used to infer a state vector containing information on atmospheric properties like CO<sub>2</sub> concentration on 20 pressure levels, surface pressure, temperature, and aerosol optical depth (AOD). The state vector also includes surface properties like albedo and solar-induced chlorophyll fluorescence (SIF). The primary scalar quantity of interest is the column-averaged CO<sub>2</sub> concentration (XCO<sub>2</sub>).

### 3.2 Atmospheric radiative transfer

A key part to inferring XCO<sub>2</sub> from observed radiances is the construction of a computational atmospheric radiative transfer model which describes how solar radiation is propagated, reflected, and scattered by Earth's surface and atmosphere. Together with an instrument model, this computer code is known as the full-physics (FP) model, referred to in this work as

$$\mathbf{y} = F(\mathbf{x}, \mathbf{b}), \quad (8)$$

where  $\mathbf{y}$  is the output of the FP model (a wavelength-by-wavelength radiance),  $\mathbf{x}$  is a state vector containing atmospheric and surface information, and  $\mathbf{b}$  denotes model parameters held fixed during data processing. A thorough description of the FP model is given in the ATBD (Boesch et al., 2015). To motivate our emulation approach, we will here describe parts of the forward model physics, which is not intended to be a full description of the included physics. Rather, we leverage this information to better design our emulator.

Part of the radiance comes from absorption of radiation by atmospheric molecules, given by

$$I(\lambda) = f_0(\lambda) \cos(\tau_0) \cdot R(\lambda, \theta, \theta_0, \varphi - \varphi_0) \exp(-g(\lambda)), \quad (9)$$

where  $\lambda$  is wavelength, the  $j$ th wavelength corresponds to the  $j$ th entry of radiance  $y$ ,  $f_0(\lambda)$  is the solar flux at the top of the atmosphere,  $R(\lambda, \theta, \theta_0, \varphi - \varphi_0)$  is the reflectance of the surface,  $g(\lambda)$  is an integral over radiation path length that sums

over for all modeled absorbers,  $\theta$  and  $\varphi$  are the observation zenith and azimuth angles, and  $\theta_0$  and  $\varphi_0$  are the corresponding solar zenith and azimuth angles. Observation and solar angles have a significant effect on the observed and modeled radiances, which will be important later in this work.

After calculating the absorption with Eq. (9), equations further describing atmospheric scattering are employed to solve for *atmospheric radiative transfer* (RT), which describes the total effect of atmosphere and surface on the scattered photons. The FP framework further includes an instrument model, which describes the effects of the observing system to the *top-of-the-atmosphere radiances*. These effects include instrument Doppler shift, spectral dispersion, and convolution with the instrument line shape (ILS) function, reducing the resolution from the finer RT grid to the coarser observational grid. On an abstracted level, this corresponds mathematically to

$$I_{\text{OBS}}(\lambda) = C_1(\lambda) \int_{-\infty}^{+\infty} \text{RT}(\lambda') \text{ILS}(\lambda, \lambda') d\lambda' + C_2(\lambda), \quad (10)$$

where  $C_1(\lambda)$  and  $C_2(\lambda)$  denote the instrument effects other than convolution that can be expressed as multiplication and addition. Generally speaking, the instrument effects depend on different physical properties that can vary between detector arrays, while the RT portion of the forward model is constant within the instrument. This observation motivates us to focus on emulating the outputs of the RT, referred to as *monochromatic radiances*, after which instrument functions can be applied appropriately after the fact. Looking forward to operational integration of our emulator, this will reduce the complexity of the emulated system and arguably make our task easier.

### 3.3 OCO-2 state vector

The state vector elements comprising  $\mathbf{x}$  for the FP model are summarized in Table 1. Notably, we have divided the table into two parts. The upper half lists the previously mentioned atmospheric and surface state vector elements that affect the RT part only, and the rest having to do with the instrument effects are in the lower half. This collection includes scaling factors for empirical orthogonal functions (EOFs) that capture unmodeled offsets in the observed radiances (O'Dell et al., 2018).

In addition to state vector elements, the FP model is parameterized by a set of parameters that are held fixed based on auxiliary information, such as laboratory measurements or meteorological data sets. These parameters include instrument calibration details, spectroscopy properties for absorbing gases, land elevation, and aerosol microphysical parameters. These aerosol parameters arise from the selection of two dominant aerosol types as a function of space and time. All aerosol types have different optical properties. This choice is determined a priori by global maps based on meteorolog-

**Table 1.** Elements of the OCO-2 state vector by functional group. The second column indicates the total elements per group. The check marks in the remaining columns indicate which wavelength bands are sensitive to changes in each variable.

State vector element	No. elements	O <sub>2</sub>	WCO <sub>2</sub>	SCO <sub>2</sub>
CO <sub>2</sub> concentration profile	20		✓	✓
H <sub>2</sub> O scaling factor	1	✓	✓	✓
Surface pressure (pascals)	1	✓	✓	✓
Temperature offset (kelvin)	1	✓	✓	✓
Aerosol height, width, and AOD	12	✓	✓	✓
O <sub>2</sub> band albedo	2	✓		
WCO <sub>2</sub> band albedo	2		✓	
SCO <sub>2</sub> band albedo	2			✓
O <sub>2</sub> band dispersion	2	✓		
WCO <sub>2</sub> band dispersion	2		✓	
SCO <sub>2</sub> band dispersion	2			✓
O <sub>2</sub> band EOF scaling	3	✓		
WCO <sub>2</sub> band EOF scaling	3		✓	
SCO <sub>2</sub> band EOF scaling	3			✓
SIF parameters	2	✓		

ical knowledge and measurements (see Fig. 1). The possible dominant aerosol types are dust (DU), sulfate (SO), sea salt (SS), organic carbon (OC), and black carbon (BC). While constructing the emulator, we will consider data sets with a fixed pair of dominant aerosol species in order to decouple their physical effects from the rest of state vector. Separate emulators can then be constructed for each pair of aerosol species, and a selection of which one to use can be done by matching the measurement location with the appropriate types.

### 3.4 ReFRACtor

This work develops a proof-of-concept version of OCO-2 forward model emulator for a simulated case. For this reason and ease of access, we implement our simulations using the Reusable Framework for Atmospheric Composition (ReFRACtor, McDuffie et al., 2020). ReFRACtor is an extensible multi-instrument atmospheric composition retrieval framework that supports and facilitates combined use of radiance measurements from different instruments in the ultraviolet, visible, near-infrared, and thermal-infrared. It has been open-source since 2014 when it was first developed as the Level-2 processing code for OCO-2. Since 2017 the development team has worked to create a more general framework that supports more instruments and spectral regions. This framework has been developed to provide the broader Earth science community a freely licensed software package that uses robust software engineering practices with well-tested, community-accepted algorithms and techniques. ReFRACtor is geared not only towards the creation of end-to-end production science data systems, but also towards scientists who need a software package to help investigate specific Earth science atmospheric composition questions. Although

ReFRACtor includes an implementation of a version of the OCO-2 production algorithm, the two have drifted since the initial intercomparison work was done. At that time it was validated against the B9.2.00 version of the software. For the most part mainly bug fixes have been kept in sync between the two versions. Additionally the core radiative transfer algorithms are the same, which justifies the use of ReFRACtor for constructing our emulator at this stage. Some minor additional algorithmic features made their way into the ReFRACtor version of OCO-2 from the production version. For the most part the major discrepancy will be due to changes in configuration values not implemented in ReFRACtor. These include values such as a priori and covariance versions, EOF data sets, ABSCO versions, and the solar model.

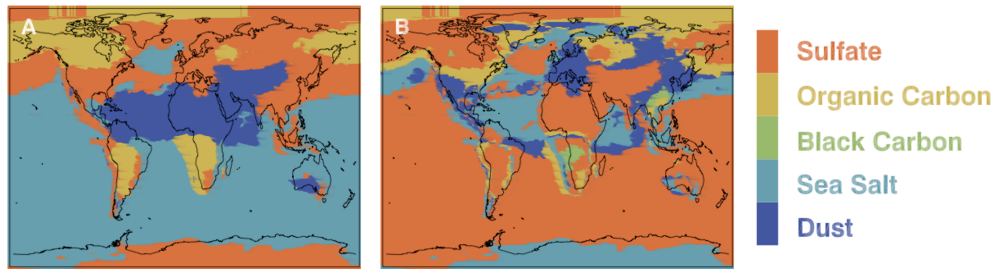
### 3.5 Retrieval algorithm

Inferring XCO<sub>2</sub> from measured radiances is an ill-posed inverse problem, which is referred to as performing a retrieval. The relationship between measurement and state is first modeled as

$$\mathbf{y} = F(\mathbf{x}) + \boldsymbol{\varepsilon}, \quad (11)$$

where data  $\mathbf{y} \in \mathbb{R}^n$  are a radiance vector; unknown  $\mathbf{x} \in \mathbb{R}^m$  is the state vector,  $F: \mathbb{R}^m \rightarrow \mathbb{R}^n$  is the OCO-2 FP model; and  $\boldsymbol{\varepsilon} \in \mathbb{R}^n$  is the measurement uncertainty. For completeness, we summarize the operational retrieval algorithm used in OCO-2 processing. The retrieval proceeds with solving the inverse problem by using Bayesian formulation, in which the additive error  $\boldsymbol{\varepsilon}$  and the prior for  $\mathbf{x}$  are assumed to be Gaussian such that

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \mathbf{S}_\varepsilon), \quad \mathbf{x} \sim \mathcal{N}(\mathbf{x}_a, \mathbf{S}_a). \quad (12)$$



**Figure 1.** Example global map of (a) primary and (b) secondary aerosol types used in the OCO-2 FP model. Different aerosol types imply different physics, which needs to be taken into account when building a forward model emulator. Image taken from Boesch et al. (2015).

The measurement error covariance matrix  $\mathbf{S}_\varepsilon$  is assumed to be diagonal, with elements for each wavelength  $j$  given by

$$\sigma_j^2 = k_1 y_j + k_2, \quad (13)$$

where  $k_1$  and  $k_2$  are calibration parameters adjusted by the instrument calibration team. The a priori covariance is taken to be diagonal for non- $\text{CO}_2$  parameters, and the  $\text{CO}_2$  profile is assumed to have a correlation structure shown in Fig. 2, which promotes continuous concentration profiles and limits the variability higher up in the atmosphere.

The retrieval is operationally carried out using iterative gradient-based methods to solve for the maximum a posteriori estimate, which is equivalent to minimizing the cost function:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} (\mathbf{y} - F(\mathbf{x}))^T \mathbf{S}_\varepsilon^{-1} (\mathbf{y} - F(\mathbf{x})) + (\mathbf{x} - \mathbf{x}_a) \mathbf{S}_a^{-1} (\mathbf{x} - \mathbf{x}_a). \quad (14)$$

This optimization problem is solved using the Levenberg–Marquardt algorithm, in which at iteration  $i$  the state is updated according to

$$\begin{aligned} & \left( (1 + \gamma) \mathbf{S}_a^{-1} + \mathbf{K}_i^T \mathbf{S}_\varepsilon^{-1} \mathbf{K}_i \right) d\mathbf{x}_{i+1} \\ & = \left[ \mathbf{K}_i^T \mathbf{S}_\varepsilon^{-1} (\mathbf{y} - F(\mathbf{x}_i)) + \mathbf{S}_a^{-1} (\mathbf{x}_a - \mathbf{x}_i) \right], \end{aligned} \quad (15)$$

where  $\gamma$  is a damping parameter and  $\mathbf{K}_i$  is the Jacobian of  $F(\mathbf{x})$  at iteration  $i$ . After each iteration, before updating the state, the effect of forward model nonlinearity is assessed by computing the quantity

$$R = \frac{c_i - c_{i+1}}{c_i - c_{\text{FC}}}, \quad (16)$$

where  $c_i$  is the value of the cost function (14) at iteration  $i$ ,  $c_{i+1}$  is similarly at iteration  $i + 1$ , and  $c_{\text{FC}}$  is the cost function value assuming that  $F(\mathbf{x}_i + d\mathbf{x}_{i+1}) = F(\mathbf{x}_i) + \mathbf{K}_i d\mathbf{x}_{i+1}$  (i.e., a linear update). Based on the value of  $R$ , one of the following is executed:

- $R \leq 0.0001$ :  $\gamma$  is increased by a factor of 10. State is not updated.

- $0.0001 < R < 0.25$ :  $\gamma$  is increased by a factor of 10;  $\mathbf{x}_{i+1} = \mathbf{x}_i + d\mathbf{x}_{i+1}$ .

- $0.25 < R < 0.75$ :  $\mathbf{x}_{i+1} = \mathbf{x}_i + d\mathbf{x}_{i+1}$ .

- $R > 0.75$ :  $\gamma$  is decreased by a factor of 2,  $\mathbf{x}_{i+1} = \mathbf{x}_i + d\mathbf{x}_{i+1}$ .

After each nondivergent step, convergence is assessed by computing the error variance derivative (see Boesch et al., 2015, for details). The operational retrieval further provides an estimate for the posterior covariance as a Laplace approximation:

$$\hat{\mathbf{S}} = \left( \mathbf{K}^T \mathbf{S}_\varepsilon^{-1} \mathbf{K} + \mathbf{S}_a^{-1} \right)^{-1}. \quad (17)$$

This is done together with the so-called *averaging kernel*:

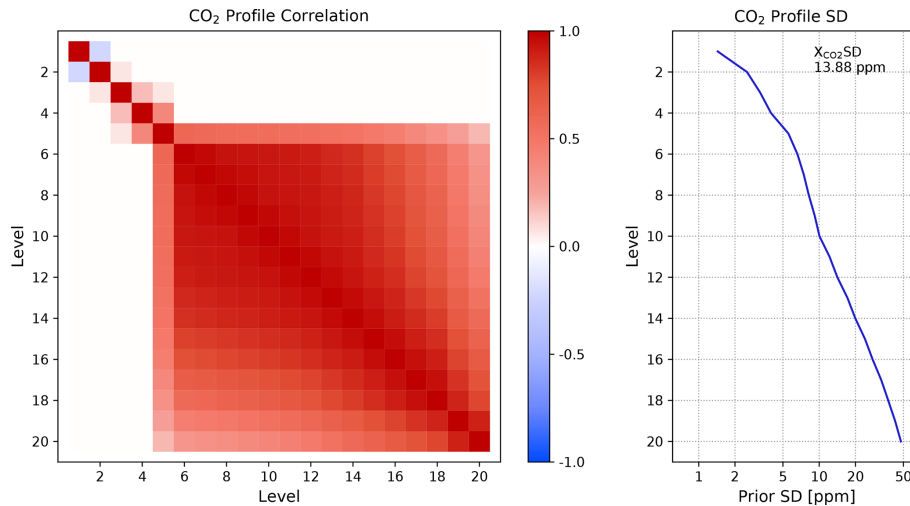
$$\mathbf{A} = (\mathbf{S}_a^{-1} + \mathbf{K}^T \mathbf{S}_\varepsilon^{-1} \mathbf{K})^{-1} \mathbf{K}^T \mathbf{S}_\varepsilon^{-1} \mathbf{K}, \quad (18)$$

which can be interpreted as the sensitivity of the retrieved state  $\hat{\mathbf{x}}$  to the true atmospheric state  $\mathbf{x}$ . These quantities are important for downstream users of OCO data products, which highlights the value of producing closed-form Jacobians during data processing.

#### 4 Forward model emulation

In this section, we will describe the practical implementation of our method laid out in Sect. 2 applied to the OCO-2 retrieval problem in Sect. 3. This includes data transformations and dimension reduction, training data generation, convergence of the optimizer in kernel parameter learning, and assessment of forward model output quality. We stress that in order to be implemented in an operational retrieval algorithm, the emulator is required to perform with superior accuracy. We ensure accurate performance by making sure that the error in predicted radiances, compared to FP outputs, is less than the radiance measurement error standard deviation. This way, any systematic errors in emulation will be masked by measurement noise, and retrieval performance using emulation will closely resemble that of using the FP model.





**Figure 2.** The a priori correlation matrix and standard deviation used for the CO<sub>2</sub> vertical profile in the OCO-2 retrieval. Vertical levels are ordered from the top of the atmosphere (Level 1) to the surface (Level 20).

#### 4.1 Data transformations

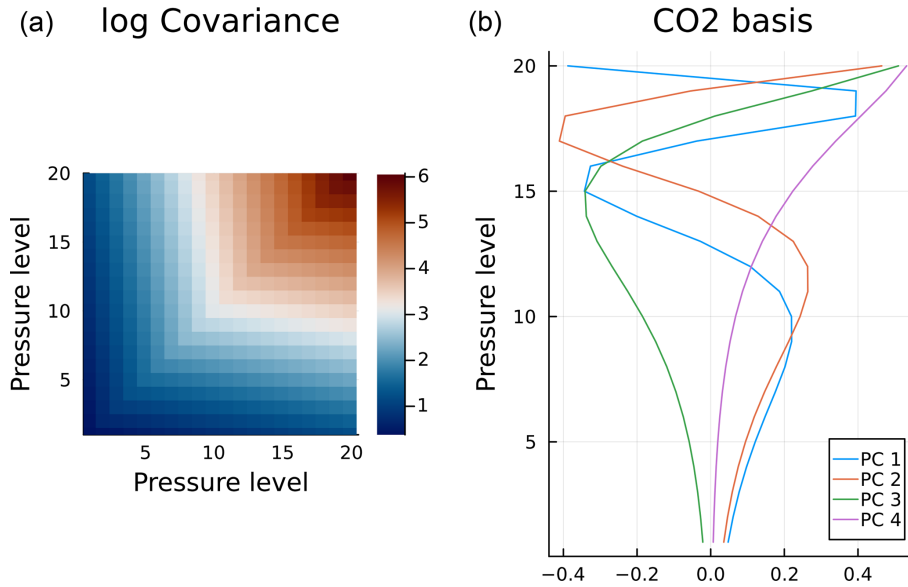
As GPs tend to perform worse with increasing input dimension and because the standard GP formulation is developed for one-dimensional outputs, we will need to reduce the dimension of both the atmospheric state and the radiance. For the atmospheric state  $\mathbf{x}$ , we leverage the fact that OCO-2 measurements are made at three separate wavelength bands, which leads to the state vector having band-specific elements, which can be ignored when dealing with other bands. This partition has been summarized in Table 1. Earlier work by Ma et al. (2019) considered cross-band correlations while emulating OCO-2 radiances, but the authors finally showed that the bands are distant enough from one another in wavelength space that they can be treated independently. With this insight, we proceed by constructing separate GPs for each band and using only the sensitive dimensions of  $\mathbf{x}$  as inputs. We further notice that the 20-element CO<sub>2</sub> profile is continuous and can be expressed as loadings obtained using principal component analysis (PCA). The most straightforward way to do this is by truncated singular value decomposition (SVD) of the empirical covariance matrix of state vectors (Tukiainen et al., 2016). To accomplish this, we use a simulation distribution derived by Braverman et al. (2021) for one selected set of realistic geophysical conditions as a basis for our experiments and perform SVD on the covariance matrix of this distribution. Analysis of singular value decay suggests that the CO<sub>2</sub> profile can be represented with just four principal components, which we collect to a matrix  $\mathbf{P}_x$  as the four leading singular vectors. We then project the CO<sub>2</sub> profile to a principal component space and further standardize the states by using the mean and variance of the simulation

distribution, leading to

$$\tilde{\mathbf{x}} = \frac{1}{\sigma_x} \left( \text{diag}(\mathbf{P}_x^T, \mathbb{I}_c) (\mathbf{x} - [\boldsymbol{\mu}_p, \mathbf{0}_c]) - \boldsymbol{\mu}_x \right), \quad (19)$$

where  $\boldsymbol{\mu}_p$  is the CO<sub>2</sub> profile mean,  $\boldsymbol{\mu}_x$  and  $\sigma_x$  are the state mean and state standard deviation,  $\text{diag}(A, B)$  denotes a block diagonal matrix with blocks  $A$  and  $B$ ,  $\mathbb{I}_c$  is a  $c = m - 16$ -dimensional identity matrix (as the profile is represented by 4 dimensions instead of original 20), and  $[\boldsymbol{\mu}_p, \mathbf{0}_c]$  is a stacked vector of CO<sub>2</sub> profile mean and a  $c$ -dimensional zero vector.

Next, we generate training data using a Sobol' sequence (see Sect. 2.4). For this study, we can omit dispersion, EOF and SIF parts of the state vector (see Table 1), and fix them to the prior mean. This follows from the discussion in Sect. 3 focusing on monochromatic radiances. Omitting dispersion simplifies computations as the wavelength grid would otherwise shift, making SVD for radiance dimension reduction hard. Ma et al. (2019) solved this problem by employing functional principal component analysis, while we can proceed with ordinary SVD. The empirical orthogonal functions (EOFs) are included in the operational retrieval to reduce fit residuals and therefore make convergence analysis easier. These have no direct impact on our study and can be safely omitted. Furthermore, the SIF parameters are fit on the O<sub>2</sub> band only as part of the instrument effects, and we do not include them in the emulation for this reason. As is evident from Eq. (9), the measurement geometry has a significant impact on the output of the FP model. For this reason we include three extra parameters,  $\theta$ ,  $\theta_0$ , and  $\varphi - \varphi_0$ , to our training data vector. Sufficient and realistic limits to these parameters are obtained from the simulation distribution of Braverman et al. (2021) by considering a  $4\sigma$  interval around the mean values. In all, we now have  $m = 4 + 21 + 3 = 28$  for input space, coming from profile PCs, other included state



**Figure 3.** (a) The CO<sub>2</sub> profile covariance matrix of the simulation distribution used in this work. (b) Four leading singular vectors from the SVD of the covariance matrix.

vector elements, and geometry. We create a Sobol' sequence of 20 000 points for training and scale all dimensions of the hypercube to  $[-4, 4]$ , corresponding to 4 standard deviations on the normalized  $\tilde{\mathbf{x}}$  basis. We further obtain the training data set in original space by reversing the transformation (19).

Training data  $\mathbf{Y}$  (radiances) are obtained by evaluating the FP model on each  $\mathbf{x}$  from the scaled Sobol' sequence. For this work, we choose a single realistic land nadir measurement to represent physical parameters not included in state vector  $\mathbf{x}$ . We perturb sampling geometry to reflect relevant solar and instrument angles. For a real-world application this approach can be extended to include different scenes and other location-dependent parameters. To obtain the labels  $\mathbf{z}$ , we similarly perform truncated SVD on the radiances  $\mathbf{Y}$  separately on each wavelength band  $B \in [\text{O}_2, \text{WCO}_2, \text{SCO}_2]$  and collect the leading  $n_B$  singular vectors in matrices  $\mathbf{P}_B$ . The four leading singular vectors for each band are included in Fig. 4 to present the kinds of features the most significant principal components, or basis vectors, encode. While this decomposition could hold additional information on physical processes behind the radiative transfer model, we do not pursue such an analysis further in this work. With additional standardization of the variables, we obtain the following transformations for each wavelength band  $B$ :

$$\tilde{\mathbf{z}}_B = \frac{1}{\sigma_z} \left( \mathbf{P}_B^T (\mathbf{y}_B - \boldsymbol{\mu}_B) - \boldsymbol{\mu}_z \right), \quad (20)$$

where  $\boldsymbol{\mu}_B$  is the radiance mean, and  $\boldsymbol{\mu}_z$  and  $\sigma_z$  are the principal component mean and standard deviation for band  $B$ .

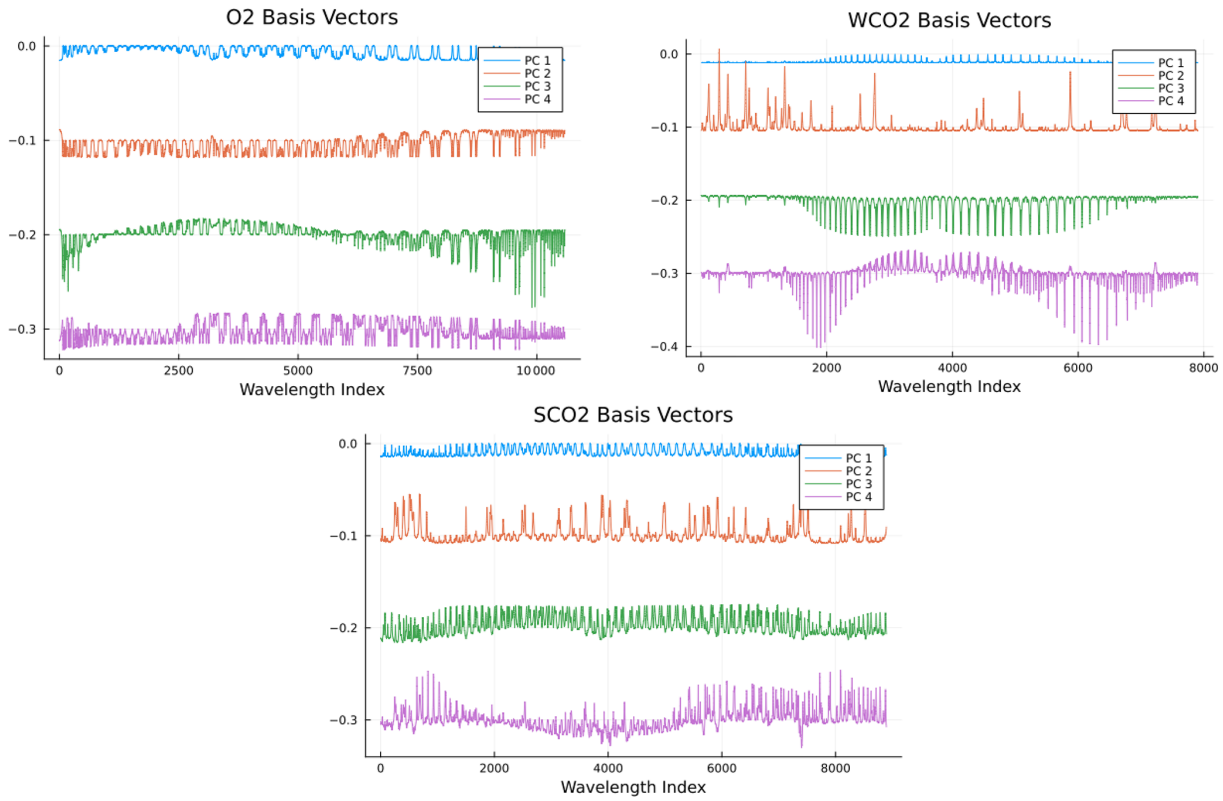
The quality of this approximation is assessed by plotting the reconstruction  $\mathbf{P}_B \mathbf{P}_B^T (\mathbf{y}_B - \boldsymbol{\mu}_B) + \boldsymbol{\mu}_B$  over a holdout data set not used in computing the SVD. We illustrate in the up-

per panel of Fig. 5 the distribution of relative reconstruction error from this data set. We have further applied the instrument function to each residual and further divided them by the measurement error standard deviation given by Eq. (13). This metric is justified by the rationale that if the reconstruction error is less than or comparable to measurement error on the radiances, no significant amount of information is lost.

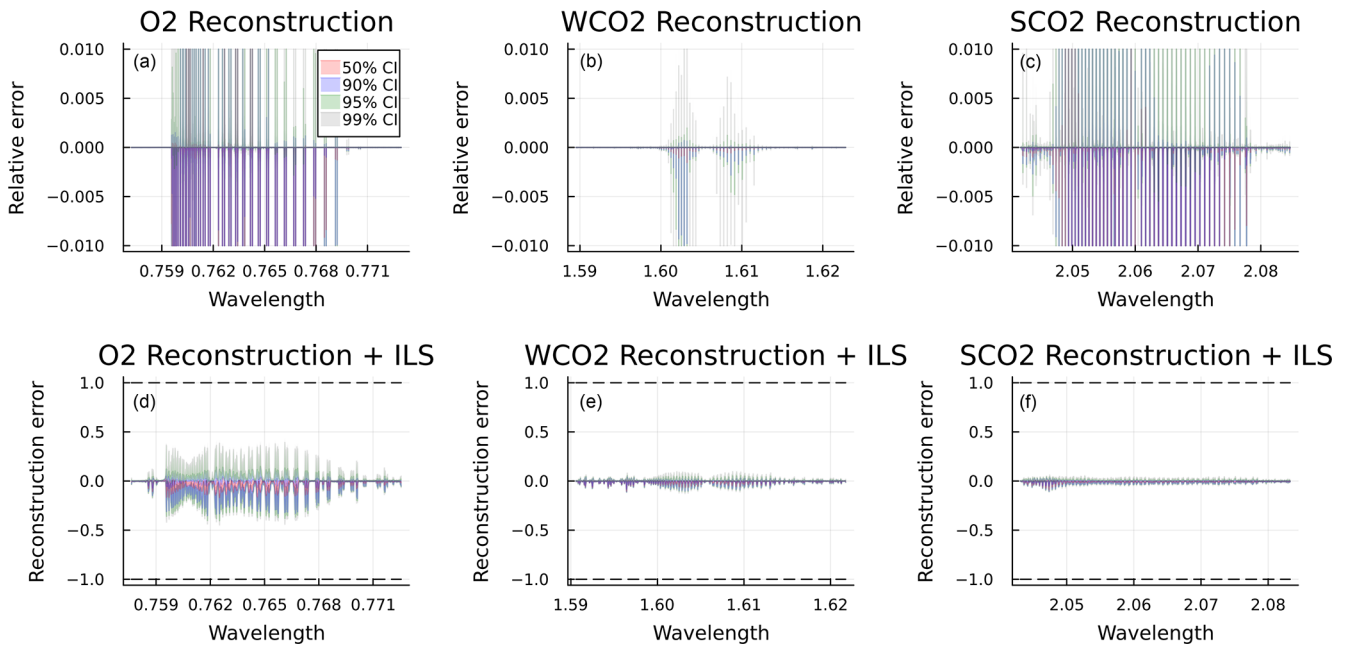
The final emulator  $g(\mathbf{x})$  can now be summarized in Fig. 6, where  $\text{GP}_i(\tilde{\mathbf{x}}_B)$  is the GP prediction given by Eq. (1), and the indices  $i$ ,  $j$ , and  $k$  run through the number of principal components included in a given band. The effect of this choice will be examined further later in this work. The state is first normalized according to Eq. (19) for each band  $B$ , after which the GP-predicted principal component loadings are assembled back to radiances using relation in Eq. (20). When evaluating the emulator, each index is independent and can then be computed in parallel.

## 4.2 Training

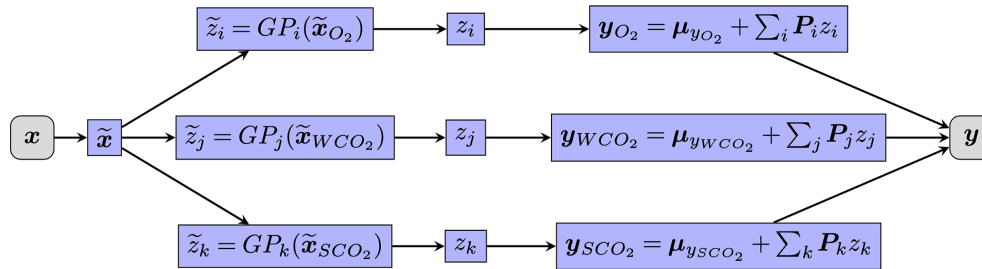
Having obtained training data  $\tilde{\mathbf{X}}$  and  $\mathbf{z}$ , we can now proceed in optimizing the kernel parameters as described in Sect. 2.3. We prescribe an individual GP per output parameter  $z_i$ . We have  $N = 20\,000$  for training data size, and we set  $M = 100$  for mini-batch size, set  $p = 5$  for the number of prediction points per mini-batch, and run the ADAM optimizer for 5000 iterations with a small learning rate – in our case 0.02. We initialize all other parameters at 1, except for linear component weight at 0 and the nugget at  $1 \times 10^{-6}$  for  $\tilde{\mathbf{x}}$ . As outlined in Sect. 3.3, we further reduce the dimension of the input space by selecting only the indices that a given wavelength band is sensitive to, given by Table 1.



**Figure 4.** Leading four basis vectors obtained from the SVD (principal components, PCs) for radiances  $y$  for each band: O<sub>2</sub>, weak CO<sub>2</sub> (WCO<sub>2</sub>), and strong CO<sub>2</sub> (SCO<sub>2</sub>). Basis vectors 2–4 are offset for illustration purposes.



**Figure 5.** (a, b, c) Distribution of relative reconstruction error for monochromatic radiances on the O<sub>2</sub>, WCO<sub>2</sub>, and SCO<sub>2</sub> bands. (d, e, f) Distribution of reconstruction error for all bands after applying the instrument function and dividing by measurement error standard deviation. Shading represents 50 % (red), 90 % (blue), 95 % (green), and 99 % (gray) confidence intervals.



**Figure 6.** Diagram showing the step-by-step process of emulator evaluation. Transformation  $\tilde{x}$  refers to normalization in Eq. (19); similarly  $\tilde{z}$  denotes the scaling given in Eq. (20).  $GP(\tilde{x})$  denotes the GP prediction given by Eq. (1) of labels  $z$ , which are assembled back to radiances based on Eq. (20) for each band and each principal component  $i$ ,  $j$ , and  $k$  therein.

For testing the performance of the algorithm, we draw a random sample  $\mathbf{X}^{\text{test}}$  from the same simulation distribution from Braverman et al. (2021) as independent test data, which is then used to evaluate the FP model to create radiances  $\mathbf{Y}^{\text{test}}$ . For test data, we fix dispersion, EOFs, and SIF at prior values as before. Example behavior of the loss function together with evolution of the kernel parameter values and true vs. predicted  $z$  values is shown in Fig. 7. We see that during training, the loss function values converge to a small value close to 0. The fluctuations of loss function values are due to the ADAM optimizer effectively being a stochastic gradient descent method: each mini-batch is sampled randomly, which causes consequent steps, possibly resulting in higher values, while the running average still always decreases. We can also see the evolution of different kernel parameters: the weight of the linear component, for example, was initially set very close to 0. During training, the algorithm correctly identifies the significance of the linear term, and the relative importance of this term overtakes the nugget (corresponding to random noise). Resulting predictions of principal component loadings can be seen to land very tightly on the one-to-one line, indicating good performance over the whole test data. The distribution of true vs. predicted  $z$  values for each component on each wavelength band is further illustrated in Fig. 8. We conclude that the predicted values correspond the true values most of the time. Some principal components show a larger spread on prediction errors (e.g., O<sub>2</sub> PC 9 or WCO<sub>2</sub> PC 9). These principal components can be redundant and do not contain meaningful information about the radiance, since the total predictive performance still remains very accurate.

### 4.3 Predictive performance

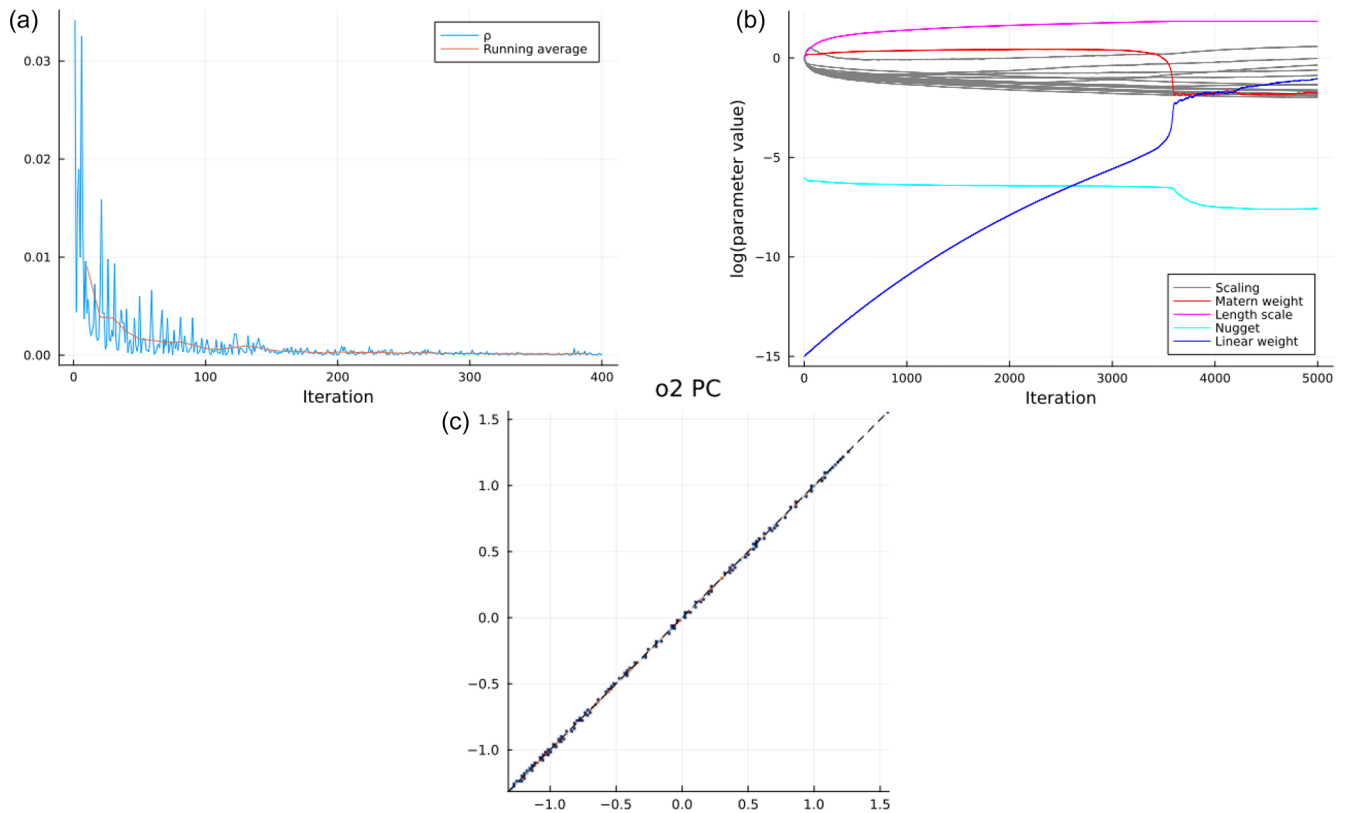
Finally, we assemble the predicted  $z$  values back to radiances and compute the relative differences with the test data, shown in the upper panel of Fig. 9. On the lower panels, as before, we apply the instrument function to these residuals and divide by the measurement error standard deviation to underline that the desired performance would be to make less prediction error than measurement error.

**Table 2.** Evaluation times of the radiative transfer (RT) model and related Jacobian, comparing the ReFRACtor implementation, monochromatic emulator with instrument line shape (ILS) and other spectral corrections, and monochromatic emulator only.

	RT [s]	RT + Jacobian [s]
ReFRACtor	33.45	55.26
Emulator + ILS	2.06	2.17
Emulator	0.05	0.19

After constructing the emulator obtaining radiances as outputs, we can further apply Eq. (3) to compute the Jacobians  $\frac{d}{dx}z$ . We can then reverse the normalizing transformations on both  $\tilde{x}$  and  $y$  and further apply the instrument functions to our Jacobians to get back to the operational observation units. The Jacobians obtained by evaluating both the FP model and the emulator on an example state vector together with the resulting profile averaging kernels are shown in Fig. 10. Accurate averaging kernels are important for downstream usage of the retrieved XCO<sub>2</sub>, as it is used, for example, in flux inversion models to obtain the vertical sensitivity of XCO<sub>2</sub> to modeled atmospheric CO<sub>2</sub> profiles. We note that we have normalized the Jacobians and averaging kernels by maximum values of each row in the matrix for visual clarity. Although not perfectly similar, we conclude that these two outputs share significant similarity. The main difference in the averaging kernels mainly results from the choices of modeling concentration profiles by principal component loadings.

As noted in previous work by Ma et al. (2019), an emulator provides substantial appeal in terms of computational efficiency. For the current work, the average computational times for model evaluation and Jacobians are summarized in Table 2 on a 2023 MacBook Pro. It is worth mentioning that ReFRACtor computes Jacobians via automatic differentiation, while our emulator does this analytically. Three cases are contrasted: the standard ReFRACtor FP evaluation, the emulator for monochromatic radiances plus ILS, and the emulator alone.



**Figure 7.** Example training performance for the first principal component of the O<sub>2</sub> band. **(a)** Loss function values for the first 400 iterations of parameter learning. The blue line depicts the cost function value per iteration. The red line depicts a cumulative running average of the cost function. **(b)** Evolution of the kernel parameters as a function of iteration. **(c)** True vs. predicted  $z$  values over a withheld test set for the first component of the O<sub>2</sub> band.

#### 4.4 Faster research version

In recent years, the uncertainty quantification and statistics community has benefited enormously by utilizing the surrogate model by Hobbs et al. (2017) to explore the OCO-2 retrieval in numerous applications (Brynjarsdóttir et al., 2018; Lamminpää et al., 2019; Nguyen and Hobbs, 2020; Hobbs et al., 2021; Patil et al., 2022). We remark that for similar purposes, our emulator can be used as an even faster surrogate. As we see from Table 2, the majority of the computational cost for the emulator comes from the instrument effects, which are part of the ReFRACtor software. If one is not interested in including the effects of dispersion, SIF, and EOFs during the retrieval, we notice from Eq. (10) that instrument corrections to RT amount to multiplication, addition, and convolution, which is associative with respect to multiplication. We can then write the emulator as

$$g(\mathbf{x}) = \text{ILS}(\widehat{\mathbf{P}}\eta(\mathbf{x})) = \text{ILS}(\widehat{\mathbf{P}})\eta(\mathbf{x}), \quad (21)$$

where  $g(\mathbf{x})$  is the overall emulator,  $\text{ILS}()$  is a function applying the instrument corrections from Eq. (10),  $\widehat{\mathbf{P}}$  is a projection matrix consisting of radiance basis functions that correspond to transforming predicted labels  $z$  back to radiances  $\mathbf{y}$  follow-

ing the last step in Fig. 6, and  $\eta(\mathbf{x})$  is the emulator predicting labels  $z$  from inputs  $\mathbf{x}$ . Done this way, we can evaluate the instrument corrections on the basis vectors once, after which OE or MCMC can proceed an order of magnitude faster (according to Table 2).

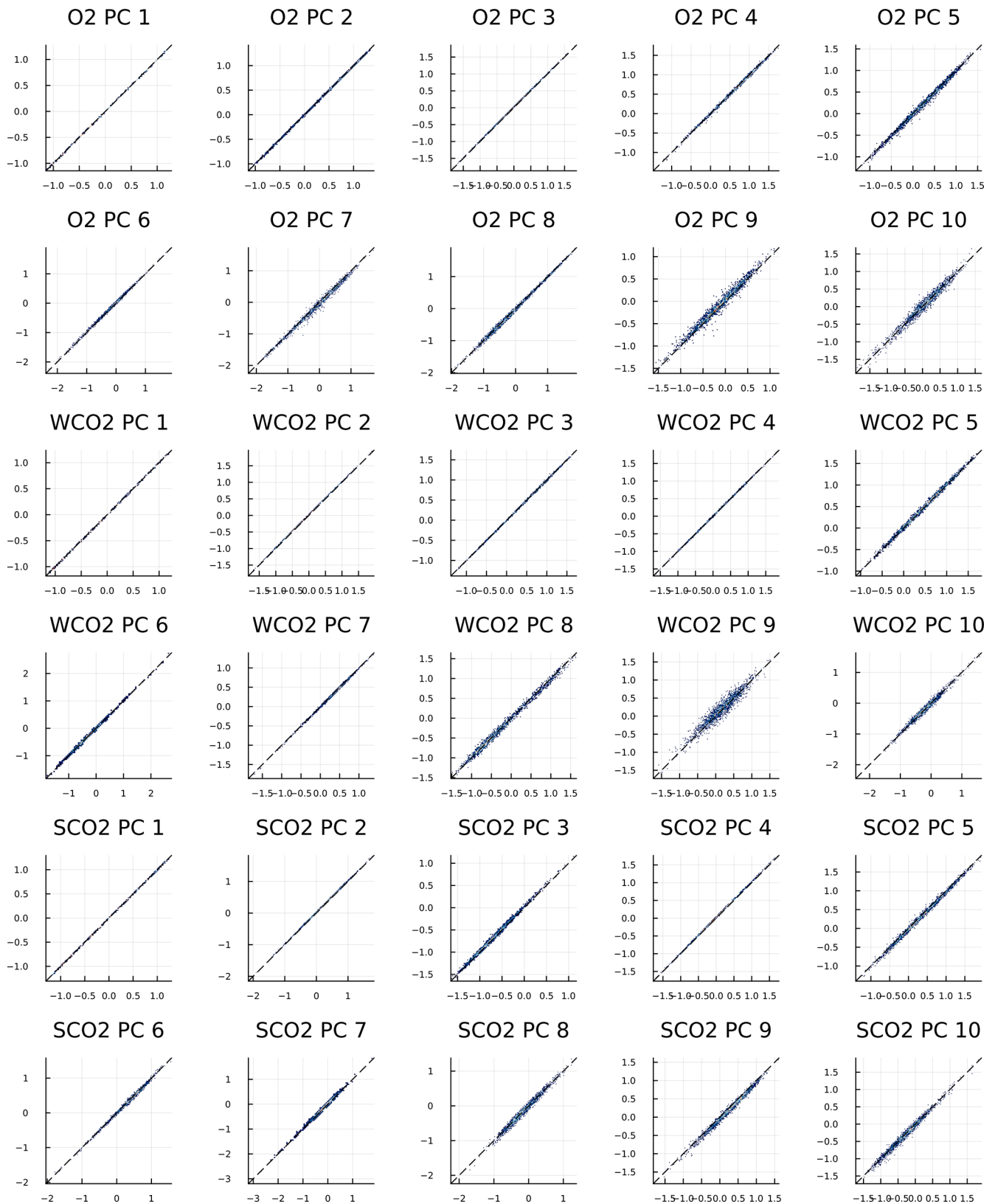
#### 5 Retrievals using the emulator

We are now ready to compare the performance of the emulator against the FP model when performing simulated retrievals. After obtaining the minimizer  $\widehat{\mathbf{x}}$  and a Laplace approximation of posterior covariance,  $\widehat{\mathbf{S}}$ , the quantity of interest is further given by multiplying the CO<sub>2</sub> profile by the pressure weighting function  $\mathbf{h}$  that puts an appropriate weight for each pressure level, resulting in

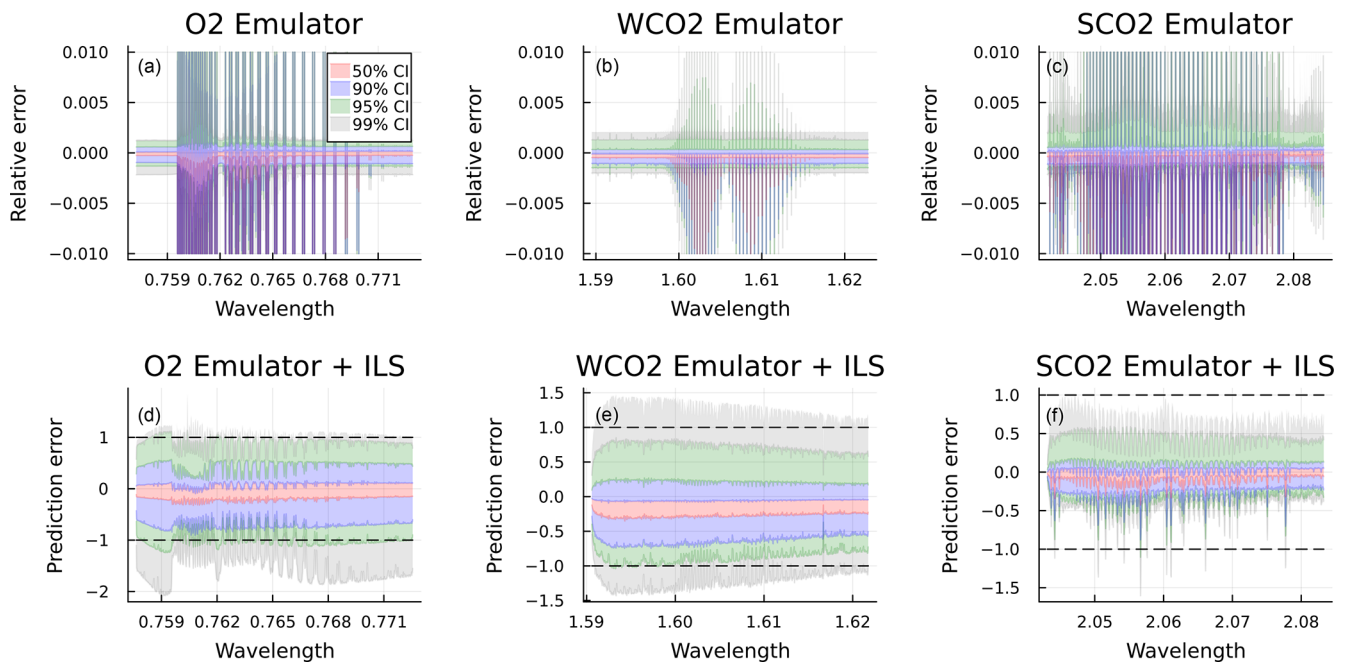
$$\text{XCO}_2 = \mathbf{h}^T \widehat{\mathbf{x}}_{1:20}. \quad (22)$$

The reported uncertainty coming with the quantity of interest (QoI) is given by

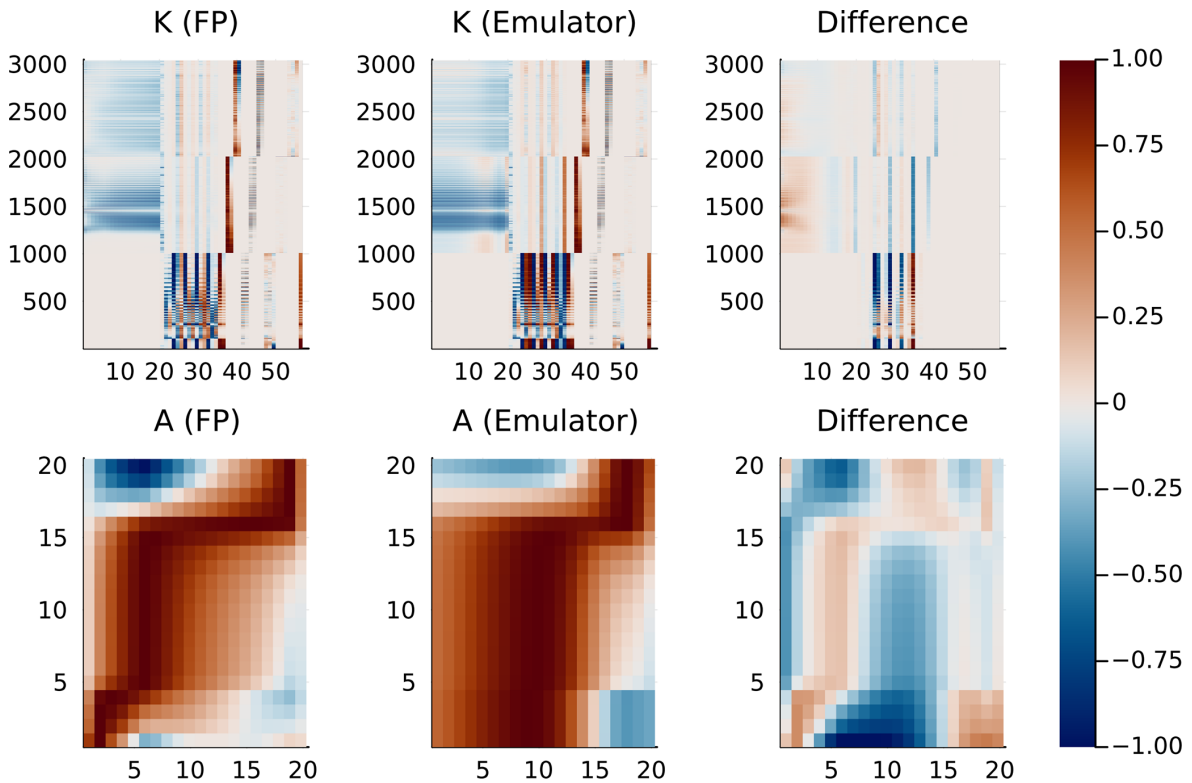
$$\text{XCO}_{2\text{uncert}} = \sqrt{\mathbf{h}^T \widehat{\mathbf{S}}_{1:20, 1:20} \mathbf{h}}. \quad (23)$$



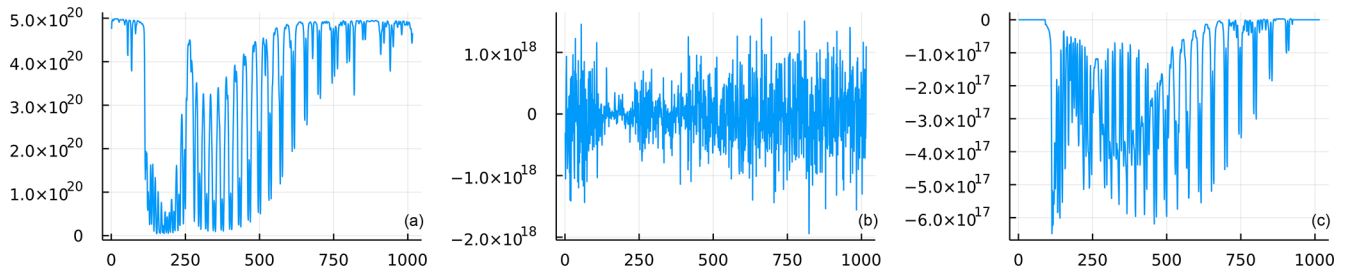
**Figure 8.** True versus predicted values for 10 radiance principal components on each wavelength band.



**Figure 9.** (a, b, c) Distribution of relative prediction error for monochromatic radiances on the O<sub>2</sub>, WCO<sub>2</sub>, and SCO<sub>2</sub> bands. (d, e, f) Distribution of the prediction error for all bands after applying the instrument function and dividing by the measurement error standard deviation. Shading represents 50 % (red), 90 % (blue), 95 % (green), and 99 % (gray) confidence intervals.



**Figure 10.** Normalized Jacobians  $\mathbf{K}$  and profile averaging kernels  $\mathbf{A}$  from both the FP model and the emulator, together with the corresponding differences.



**Figure 11.** (a) Example O<sub>2</sub> band radiance. (b) Realization from the noise distribution  $N(0, \mathbf{S}_\varepsilon)$ . (c) Realization from the model discrepancy distribution  $N(\mu_\delta, \mathbf{S}_\delta)$ . Units for all panels are  $\text{W m}^{-2} \text{sr}^{-1} \mu\text{m}^{-1}$ , the units of radiance for OCO-2.

We present two test cases for assessing retrieval performance of our emulator. First, we create synthetic observations by evaluating the FP model on our test set of states  $\mathbf{x}$  and by adding a realization from the Gaussian noise distribution:

$$\mathbf{y}_{\text{test}} = F(\mathbf{x}, \mathbf{b}) + \boldsymbol{\varepsilon}, \quad (24)$$

where  $\boldsymbol{\varepsilon} \sim N(0, \mathbf{S}_\varepsilon)$ . Second, we follow the methods outlined in Braverman et al. (2021) to further corrupt the simulated measurement by realistic *model discrepancy* (MD) adjustment, given by

$$\mathbf{y}_{\text{test}} = F(\mathbf{x}, \mathbf{b}) + \boldsymbol{\varepsilon} + \boldsymbol{\delta}, \quad (25)$$

where  $\boldsymbol{\delta} \sim N(\mu_\delta, \mathbf{S}_\delta)$ . The shape of this adjustment is illustrated in Fig. 11. As noted by the authors, model discrepancy as presented here is a statistical representation of forward modeling mismatches so that our simulated measurements would better correspond to real data.

We then perform XCO<sub>2</sub> retrievals both using the full-physics model  $F(\mathbf{x})$  and the emulator  $g(\mathbf{x})$  following the algorithm laid out in Sect. 3. Results for retrieved XCO<sub>2</sub> for both cases with and without MD are illustrated in Fig. 12. The corresponding XCO<sub>2</sub> uncertainty values are compared in Fig. 13. We conclude that using the emulator in place of the FP model in retrieval preserves the accuracy and replicates the same biases as the FP model while having a good correlation with each other. On the other hand, the output uncertainty estimates do not seem to correspond to each other, and further analysis on this output will be required in future research work.

### 5.1 Effect of PCA dimensionality

Previously in this work we did not prescribe a certain number of principal components to use in radiance dimension reduction. Figure 14 illustrates the retrieved XCO<sub>2</sub> root mean square error (RMSE) and mean absolute error (MAE) against the true known value, together with Fig. 15, illustrating radiance reconstruction and prediction RMSE and MAE similarly to Figs. 5 and 9, all as a function of the number of PCs used. We can collectively deduce that using more than

25 principal components per band does not yield any additional performance benefits. We remark that compared to the earlier work by Ma et al. (2019), who argued for one to three principal components per band, our results show that many more components are needed for accurate retrievals. This highlights the importance of empirically checking the effect of dimensionality reduction and not relying on rules of thumb such as conserving 95 % of the variability.

### 5.2 Effect of aerosol types

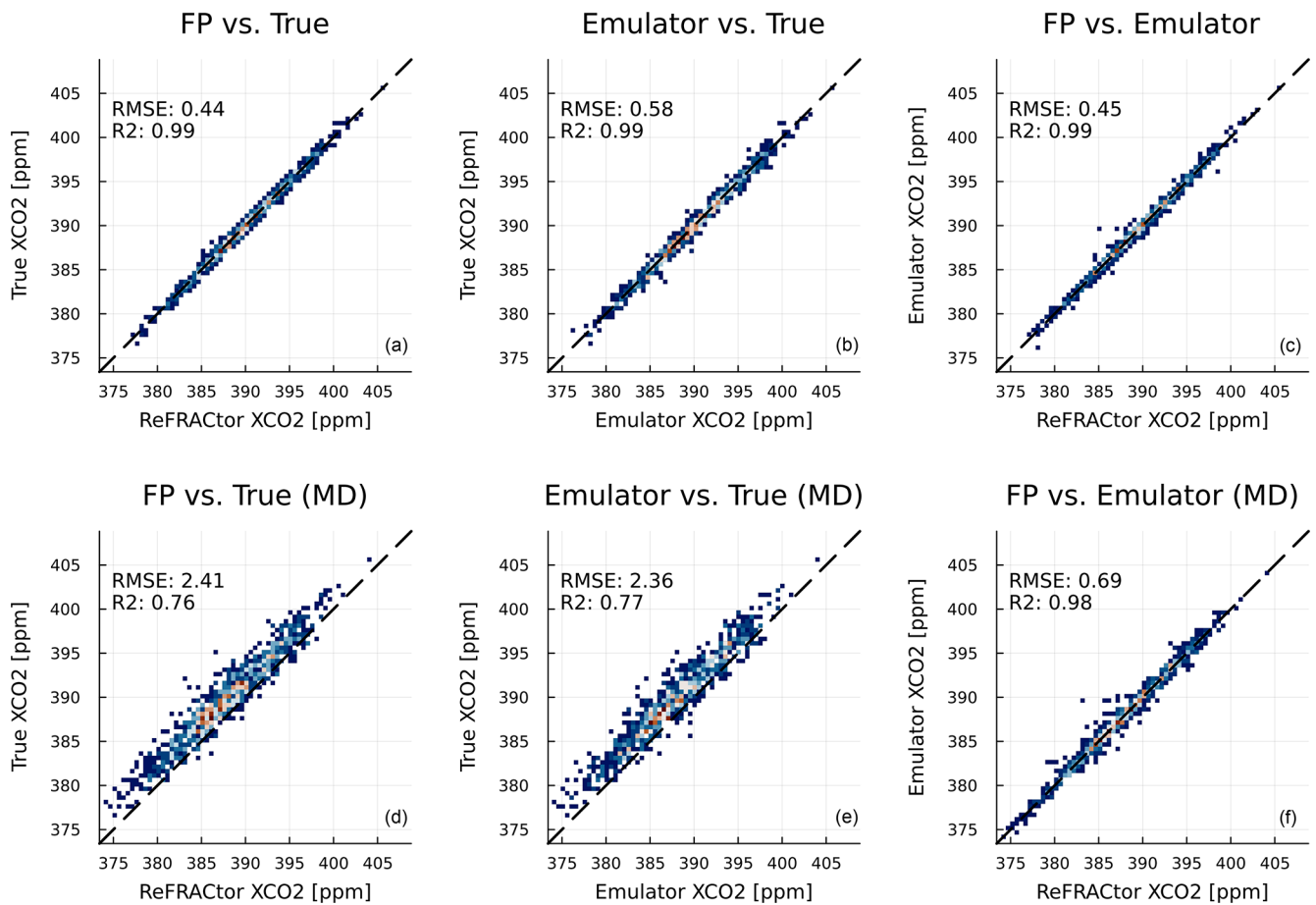
To assess the effect of changing the dominant aerosol types on the performance of the retrievals, we repeat the training and retrieval procedure described in this section with two separate pairs of dominant aerosol types. Firstly, we consider dust (DU) and sea salt (SS) and, secondly, DU and sulfate (SO). These are among the most common aerosol combinations encountered in the OCO-2 operations. We repeat the retrievals for both cases with additional MD adjustment as before. Results of this experiment are summarized in Fig. 16. We conclude that the proposed method is robust in changing physical conditions, which indicates fitness for further operational integration.

## 6 Conclusions

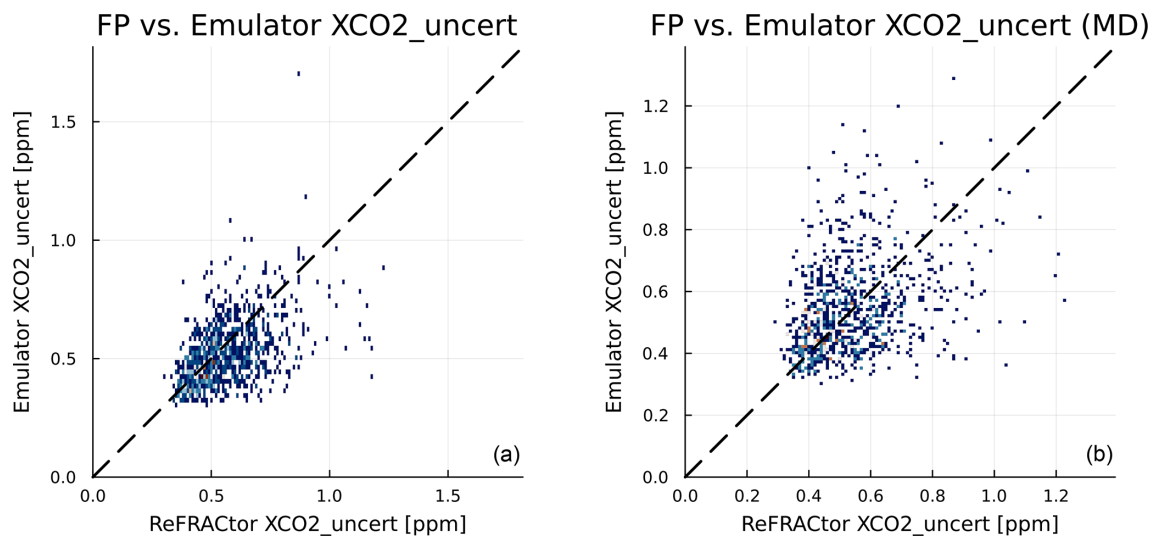
In this work, we have constructed and implemented a fast and accurate forward model emulator for the ReFRACtor implementation of the OCO-2 full-physics forward model. The emulator produces closed-form Jacobians and, as such, provides a convenient way of performing XCO<sub>2</sub> retrievals. We have demonstrated the accuracy of these retrievals and analyzed the effect of PCA dimension, aerosol types, and model discrepancy on the retrieval. All these tests indicate robustness and excellent reliability of our method and offer an encouraging proof of concept for future operational implementation with the latest ACOS algorithm and real-world OCO-2 data.

This work has significantly advanced kernel flow methodology (Owhadi and Yoo, 2019) by including a cross-validation-based training strategy using a RMSE cost function and a new strategy for mini-batching. With this method,

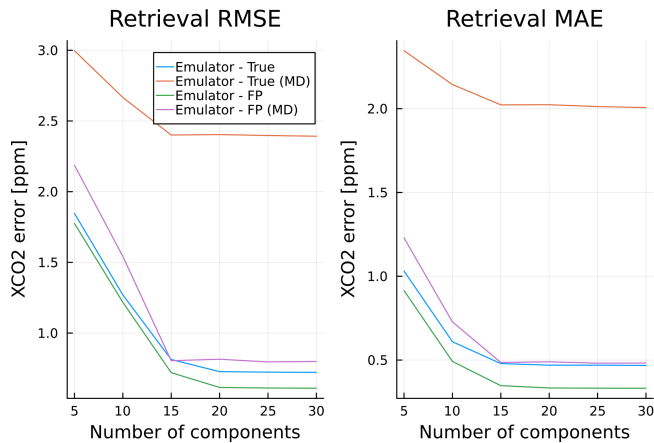




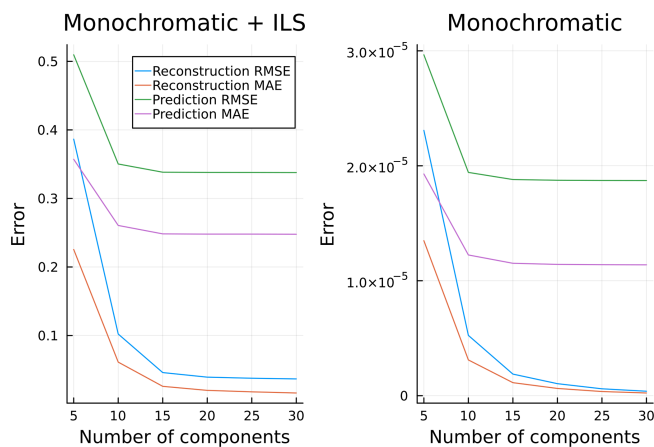
**Figure 12.** Retrieved XCO<sub>2</sub> over the holdout data set using the FP model and the emulator. Full-physics model (a), the emulator (b), and comparison of the two (c). (d, e, f) Similarly but with added model discrepancy in observed data. RMSE (root mean square error) describes the bias of the retrievals, while the  $R^2$  value is included to assess the correlation between quantities of interest.



**Figure 13.** Scatterplots of retrieval XCO<sub>2</sub> uncertainty over the holdout data set using the FP model and the emulator. (a) No MD in observations. (b) MD included in observations.



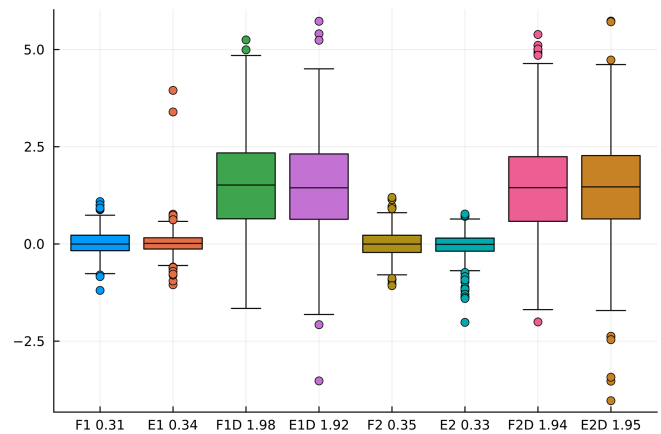
**Figure 14.** RMSE and MAE as a function of the number of principal components used per band in radiance dimension reduction for XCO<sub>2</sub> retrievals, both with and without MD, over a holdout test data set.



**Figure 15.** RMSE and MAE as a function of the number of principal components used per band in radiance dimension reduction reconstructions and predictions, all over a holdout test data set.

we have achieved a relative error of less than 1 %, which on its own is a significant improvement from the point of view of operator learning (e.g., see Battle et al., 2023, for comparisons of GP and NN methods on various nonlinear problems). Our approach is computationally fast and, when a training data set is properly engineered, performs consistently with the span of training data. Compared with our ability to compute Jacobians in closed form, our approach holds potential to solve current and future data processing issues in atmospheric remote sensing stemming from computationally intensive forward models.

While Gaussian process methods offer an attractive means of including uncertainty propagation in emulation pipeline, our tests have shown that the predicted posterior standard deviation given by GPs was not adequate in providing reliable coverage of true labels after prediction. This is likely due to



**Figure 16.** Difference (in ppm) between true and retrieved XCO<sub>2</sub> from simulated measurements with different dominant aerosol species (in ppm). Symbols on the x axis denote the specifics of a given experiment: F – full-physics model, E – emulator, 1 – DU and SS aerosols, and 2 – DU and SO aerosols. D denotes model discrepancy, followed by a number denoting mean error (in ppm).

the kernel flow method's focus on optimizing the posterior mean prediction without assessing the prediction uncertainty. This could easily be remedied by including an uncertainty tuning penalty in the kernel flow loss function. Another disclaimer comes from evaluations of retrieval uncertainty in XCO<sub>2</sub>: our method did not agree with operational OE. This does not mean our estimates were better or worse, and further research is needed in calibrating retrieval uncertainties.

A logical next step would be to implement the GP emulator for an operational ACOS forward model instead of ReFRACtor, which requires closer collaboration with the OCO algorithm team. After demonstration on OCO-2, our approach is directly applicable for a myriad of other satellite missions. We note that future work will have to deal with training data design that was simplified in this work. Assessing different temporal and spatial variability in forward model parameters together with feasible distributions of state vectors will be key in this design effort. These efforts might benefit from including a cost–benefit analysis on training a global model usable everywhere versus, for example, retraining the emulator for sufficiently specified spatiotemporal data sets.

## Appendix A: Closed-form Jacobians

To obtain a closed-form equation for the Jacobians used in the XCO<sub>2</sub> retrievals, we must explicitly compute the term  $\frac{d}{dx^*} \Gamma(x^*, \mathbf{X})$  in Eq. (3). To accomplish this, we compute the partial derivative of the kernel function (Eq. 4) with respect

to the first input:

$$\begin{aligned} & \frac{\partial}{\partial \mathbf{x}} k(\mathbf{x}, \mathbf{x}') \\ &= \frac{\partial}{\partial \mathbf{x}} \left[ \alpha_1 \left( 1 + \frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \exp \left( -\frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \right. \\ & \quad \left. + \alpha_2 (\mathcal{W}\mathbf{x})^T (\mathcal{W}\mathbf{x}') \right] \end{aligned} \quad (\text{A1})$$

$$\begin{aligned} &= \frac{\partial}{\partial \mathbf{x}} \left[ \alpha_1 \exp \left( -\frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \right] \\ & \quad + \frac{\partial}{\partial \mathbf{x}} \left[ \alpha_1 \left( \frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \exp \left( -\frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) \right] \end{aligned} \quad (\text{A2})$$

$$+ \frac{\partial}{\partial \mathbf{x}} \left[ \alpha_2 (\mathcal{W}\mathbf{x})^T (\mathcal{W}\mathbf{x}') \right] \quad (\text{A3})$$

$$\begin{aligned} &= \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \frac{\partial}{\partial \mathbf{d}} \left[ \alpha_1 \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] + \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \frac{\partial}{\partial \mathbf{d}} \left[ \alpha_1 \left( \frac{\sqrt{3}}{l} \mathbf{d} \right) \right. \\ & \quad \left. \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] + \frac{\partial}{\partial \mathbf{x}} \left[ \alpha_2 (\mathcal{W}\mathbf{x})^T (\mathcal{W}\mathbf{x}') \right] \end{aligned} \quad (\text{A4})$$

$$\begin{aligned} &= \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \left[ -\alpha_1 \left( \frac{\sqrt{3}}{l} \right) \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] + \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \left[ \alpha_1 \left( \frac{\sqrt{3}}{l} \right) \right. \\ & \quad \left. \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) - \alpha_1 \left( \frac{\sqrt{3}}{l} \right)^2 \mathbf{d} \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] \end{aligned} \quad (\text{A5})$$

$$+ \left[ \alpha_2 (\mathcal{W})^T (\mathcal{W}\mathbf{x}') \right] \quad (\text{A6})$$

$$= \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \left[ -\alpha_1 \left( \frac{\sqrt{3}}{l} \right)^2 \mathbf{d} \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] + \left[ \alpha_2 (\mathcal{W})^T (\mathcal{W}\mathbf{x}') \right] \quad (\text{A7})$$

$$\begin{aligned} &= \frac{\mathcal{W}(\mathcal{W}(\mathbf{x} - \mathbf{x}'))}{\mathbf{d}} \left[ -\alpha_1 \left( \frac{\sqrt{3}}{l} \right)^2 \mathbf{d} \exp \left( -\frac{\sqrt{3}}{l} \mathbf{d} \right) \right] \\ & \quad + \left[ \alpha_2 (\mathcal{W})^T (\mathcal{W}\mathbf{x}') \right] \end{aligned} \quad (\text{A8})$$

$$= \mathcal{W}^2 \left[ -\alpha_1 \left( \frac{\sqrt{3}}{l} \right)^2 \exp \left( -\frac{\sqrt{3}}{l} \|\mathbf{x} - \mathbf{x}'\|_{\mathcal{W}} \right) (\mathbf{x} - \mathbf{x}') + \alpha_2 \mathbf{x}' \right]. \quad (\text{A9})$$

After computing  $\frac{\partial}{\partial \mathbf{x}} k(\mathbf{x}, \mathbf{x}')$ , we get  $\frac{\text{d}}{\text{d}\mathbf{x}^*} \mathbf{\Gamma}(\mathbf{x}^*, \mathbf{X})$  element by element, with  $\mathbf{x}$  being the new input and  $\mathbf{x}'$  a training data point. The final Jacobian is then obtained by computing  $\frac{\text{d}}{\text{d}\mathbf{x}^*} z^*$  via Eq. (3) and reversing transformations in Eqs. (19) and (20).

*Code and data availability.* Code and data are available on an OSF repository at <https://doi.org/10.17605/OSF.IO/U2T8A> (Lamminpää, 2024). The software requires ReFRACtor and ReFRACtorUQ GitHub repositories, which are freely available as well.

*Author contributions.* OL: project administration, conceptualization, methodology, software, formal analysis, data curation, writing (original draft and review and editing), visualization. JS: conceptualization, methodology, software, writing (original draft and review and editing). JH: project administration, conceptualization, methodology, supervision, software, data curation, writing (review and editing). JM: methodology, software, data curation, writing (original draft). AB: conceptualization, methodology, writing (review and editing). HO: conceptualization, methodology, writing (review and editing).

*Competing interests.* The contact author has declared that none of the authors has any competing interests.

*Disclaimer.* Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. While Copernicus Publications makes every effort to include appropriate place names, the final responsibility lies with the authors.

*Acknowledgements.* The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with NASA. The authors thank Pulong Ma and Chris O'Dell for helpful guidance.

*Review statement.* This paper was edited by Peer Nowack and reviewed by two anonymous referees.

## References

- Battle, P., Darcy, M., Hosseini, B., and Owhadi, H.: Kernel Methods are Competitive for Operator Learning, arXiv [preprint], <https://doi.org/10.48550/arXiv.2304.13202>, 8 October 2023.
- Boesch, H., Brown, L., Castano, R., Christi, M., Connor, B., Crisp, D., Eldering, A., Fisher, B., Frankenberg, C., Gunson, M., Granat, R., McDuffie, J., Miller, C., Natraj, V., O'Brien, D., O'Dell, C., Osterman, G., Oyafuso, F., Payne, V., Polonski, I., Smyth, M., Spurr, R., Thompson, D., and Toon, G.: Orbiting Carbon Observatory-2 (OCO-2) Level 2 Full Physics Retrieval Algorithm Theoretical Basis, Version 2.0, Rev 2, NASA Earth Data, <https://doi.org/10.5067/8E4VLCK16O6Q>, 2015.
- Braverman, A., Hobbs, J., Teixeira, J., and Gunson, M.: Post hoc Uncertainty Quantification for Remote Sensing Observing Systems, *SIAM/ASA Journal on Uncertainty Quantification*, 9, 1064–1093, <https://doi.org/10.1137/19M1304283>, 2021.
- Bréon, F.-M., David, L., Chatelanaz, P., and Chevallier, F.: On the potential of a neural-network-based approach for estimating XCO<sub>2</sub> from OCO-2 measurements, *Atmos. Meas. Tech.*, 15, 5219–5234, <https://doi.org/10.5194/amt-15-5219-2022>, 2022.
- Brynjarsdóttir, J., Hobbs, J., Braverman, A., and Mandrake, L.: Optimal Estimation Versus MCMC for CO<sub>2</sub> Retrievals, *J. Agr.*

- Biol. Envir. St., 23, 297–316, <https://doi.org/10.1007/s13253-018-0319-8>, 2018.
- Byrne, B., Baker, D. F., Basu, S., Bertolacci, M., Bowman, K. W., Carroll, D., Chatterjee, A., Chevallier, F., Ciais, P., Cressie, N., Crisp, D., Crowell, S., Deng, F., Deng, Z., Deutscher, N. M., Dubey, M. K., Feng, S., García, O. E., Griffith, D. W. T., Herkommer, B., Hu, L., Jacobson, A. R., Janardan, R., Jeong, S., Johnson, M. S., Jones, D. B. A., Kivi, R., Liu, J., Liu, Z., Maksyutov, S., Miller, J. B., Miller, S. M., Morino, I., Notholt, J., Oda, T., O'Dell, C. W., Oh, Y.-S., Ohyama, H., Patra, P. K., Peiro, H., Petri, C., Philip, S., Pollard, D. F., Poulter, B., Remaud, M., Schuh, A., Sha, M. K., Shiomi, K., Strong, K., Sweeney, C., Té, Y., Tian, H., Velasco, V. A., Vrekoussis, M., Warneke, T., Worden, J. R., Wunch, D., Yao, Y., Yun, J., Zammit-Mangion, A., and Zeng, N.: National CO<sub>2</sub> budgets (2015–2020) inferred from atmospheric CO<sub>2</sub> observations in support of the global stocktake, *Earth Syst. Sci. Data*, 15, 963–1004, <https://doi.org/10.5194/essd-15-963-2023>, 2023.
- Chevallier, F., Ciais, P., Conway, T. J., Aalto, T., Anderson, B. E., Bousquet, P., Brunke, E. G., Ciattaglia, L., Esaki, Y., Frohlich, M., Gomez, A., Gomez-Pelaez, A. J., Haszpra, L., Krummel, P., Langenfelds, R. L., Leuenberger, M., Machida, T., Maignan, F., Matsueda, H., Morgu, J. A., Mukai, H., Nakazawa, T., Peylin, P., Ramonet, M., Rivier, L., Sawa, Y., Schmidt, M., Steele, L. P., Vay, S. A., Vermeulen, A. T., Wofsy, S., and Worthy, D.: CO<sub>2</sub> surface fluxes at grid point scale estimated from a global 21 year re-analysis of atmospheric measurements, *J. Geophys. Res.-Atmos.*, 115, D21307, <https://doi.org/10.1029/2010JD013887>, 2010.
- Connor, B. J., Boesch, H., Toon, G., Sen, B., Miller, C., and Crisp, D.: Orbiting Carbon Observatory: Inverse Method and Prospective Error Analysis, *J. Geophys. Res.*, 113, D05305, <https://doi.org/10.1029/2006JD008336>, 2008.
- Cressie, N.: *Statistics for Spatial Data*, John Wiley & Sons, Inc., <https://doi.org/10.1002/9781119115151>, 1993.
- Cressie, N.: Mission CO<sub>2</sub>ntrol: A Statistical Scientist's Role in Remote Sensing of Atmospheric Carbon Dioxide, *J. Am. Stat. Assoc.*, 113, 152–168, <https://doi.org/10.1080/01621459.2017.1419136>, 2018.
- Crisp, D., Atlas, R. M., Breon, F.-M., Brown, L. R., Burrows, J. P., Ciais, P., Connor, B. J., Doney, S. C., Fung, I. Y., Jacob, D. J., Miller, C. E., O'Brien, D., Pawson, S., Randerson, J. T., Rayner, P., Salawitch, R. J., Sander, S. P., Sen, B., Stephens, G. L., Tans, P. P., Toon, G. C., Wennberg, P. O., Wofsy, S. C., Yung, Y. L., Kuang, Z., Chudasama, B., Sprague, G., Weiss, B., Pollock, R., Kenyon, D., and Schroll, S.: The Orbiting Carbon Observatory (OCO) mission, *Adv. Space. Res.*, 34, 700–709, <https://doi.org/10.1016/j.asr.2003.08.062>, 2004.
- Crisp, D., Fisher, B. M., O'Dell, C., Frankenberg, C., Basilio, R., Bösch, H., Brown, L. R., Castano, R., Connor, B., Deutscher, N. M., Eldering, A., Griffith, D., Gunson, M., Kuze, A., Mandrake, L., McDuffie, J., Messerschmidt, J., Miller, C. E., Morino, I., Natraj, V., Notholt, J., O'Brien, D. M., Oyafuso, F., Polonsky, I., Robinson, J., Salawitch, R., Sherlock, V., Smyth, M., Suto, H., Taylor, T. E., Thompson, D. R., Wennberg, P. O., Wunch, D., and Yung, Y. L.: The ACOS CO<sub>2</sub> retrieval algorithm – Part II: Global XCO<sub>2</sub> data characterization, *Atmos. Meas. Tech.*, 5, 687–707, <https://doi.org/10.5194/amt-5-687-2012>, 2012.
- Crisp, D., Pollock, H. R., Rosenberg, R., Chapsky, L., Lee, R. A. M., Oyafuso, F. A., Frankenberg, C., O'Dell, C. W., Bruegge, C. J., Doran, G. B., Eldering, A., Fisher, B. M., Fu, D., Gunson, M. R., Mandrake, L., Osterman, G. B., Schwandner, F. M., Sun, K., Taylor, T. E., Wennberg, P. O., and Wunch, D.: The on-orbit performance of the Orbiting Carbon Observatory-2 (OCO-2) instrument and its radiometrically calibrated products, *Atmos. Meas. Tech.*, 10, 59–81, <https://doi.org/10.5194/amt-10-59-2017>, 2017.
- Crowell, S., Baker, D., Schuh, A., Basu, S., Jacobson, A. R., Chevallier, F., Liu, J., Deng, F., Feng, L., McKain, K., Chatterjee, A., Miller, J. B., Stephens, B. B., Eldering, A., Crisp, D., Schimel, D., Nassar, R., O'Dell, C. W., Oda, T., Sweeney, C., Palmer, P. I., and Jones, D. B. A.: The 2015–2016 carbon cycle as seen from OCO-2 and the global in situ network, *Atmos. Chem. Phys.*, 19, 9797–9831, <https://doi.org/10.5194/acp-19-9797-2019>, 2019.
- Datta, A., Banerjee, S., Finley, A., and Gelfand, A.: Hierarchical Nearest-Neighbor Gaussian Process Models for Large Geostatistical Datasets, *J. Am. Stat. Assoc.*, 111, 800–812, <https://doi.org/10.1080/01621459.2015.1044091>, 2016.
- David, L., Bréon, F.-M., and Chevallier, F.: XCO<sub>2</sub> estimates from the OCO-2 measurements using a neural network approach, *Atmos. Meas. Tech.*, 14, 117–132, <https://doi.org/10.5194/amt-14-117-2021>, 2021.
- Eldering, A., Taylor, T. E., O'Dell, C. W., and Pavlick, R.: The OCO-3 mission: measurement objectives and expected performance based on 1 year of simulated data, *Atmos. Meas. Tech.*, 12, 2341–2370, <https://doi.org/10.5194/amt-12-2341-2019>, 2019.
- Frey, M., Sha, M. K., Hase, F., Kiel, M., Blumenstock, T., Harig, R., Surawicz, G., Deutscher, N. M., Shiomi, K., Franklin, J. E., Bösch, H., Chen, J., Grutter, M., Ohyama, H., Sun, Y., Butz, A., Mengistu Tsidu, G., Ene, D., Wunch, D., Cao, Z., Garcia, O., Ramonet, M., Vogel, F., and Orphal, J.: Building the Collaborative Carbon Column Observing Network (COCCON): long-term stability and ensemble performance of the EM27/SUN Fourier transform spectrometer, *Atmos. Meas. Tech.*, 12, 1513–1530, <https://doi.org/10.5194/amt-12-1513-2019>, 2019.
- Friedlingstein, P., Meinshausen, M., Arora, V., Jones, C., Anav, A., Liddicoat, S., and Knutti, R.: Uncertainties in CMIP5 Climate Projections due to Carbon Cycle Feedbacks, *J. Climate*, 27, 511–526, <https://doi.org/10.1175/JCLI-D-12-00579.1>, 2014.
- Friedlingstein, P., O'Sullivan, M., Jones, M. W., Andrew, R. M., Gregor, L., Hauck, J., Le Quéré, C., Luijkx, I. T., Olsen, A., Peters, G. P., Peters, W., Pongratz, J., Schwingshackl, C., Sitch, S., Canadell, J. G., Ciais, P., Jackson, R. B., Alin, S. R., Alkama, R., Arneeth, A., Arora, V. K., Bates, N. R., Becker, M., Bellouin, N., Bittig, H. C., Bopp, L., Chevallier, F., Chini, L. P., Cronin, M., Evans, W., Falk, S., Feely, R. A., Gasser, T., Gehlen, M., Gkritzalis, T., Gloege, L., Grassi, G., Gruber, N., Gürses, Ö., Harris, I., Hefner, M., Houghton, R. A., Hurtt, G. C., Iida, Y., Ilyina, T., Jain, A. K., Jersild, A., Kadono, K., Kato, E., Kennedy, D., Klein Goldewijk, K., Knauer, J., Korsbakken, J. I., Landschützer, P., Lefèvre, N., Lindsay, K., Liu, J., Liu, Z., Marland, G., Mayot, N., McGrath, M. J., Metz, N., Monacchi, N. M., Munro, D. R., Nakaoka, S.-I., Niwa, Y., O'Brien, K., Ono, T., Palmer, P. I., Pan, N., Pierrot, D., Pöcöck, K., Poulter, B., Resplandy, L., Robertson, E., Rödenbeck, C., Rodriguez, C., Rosan, T. M., Schwinger, J., Séférian, R., Shutler, J. D., Skjelvan, I., Steinhoff, T., Sun, Q., Sutton, A. J., Sweeney, C., Takao, S., Tanhua, T., Tans, P. P., Tian, X., Tian, H., Tilbrook, B., Tsujino, H., Tubiello, F., van der Werf, G. R., Walker, A. P., Wanninkhof, R., Whitehead, C., Willstrand

- Wranne, A., Wright, R., Yuan, W., Yue, C., Yue, X., Zaehe, S., Zeng, J., and Zheng, B.: Global Carbon Budget 2022, *Earth Syst. Sci. Data*, 14, 4811–4900, <https://doi.org/10.5194/essd-14-4811-2022>, 2022.
- Gurney, K., Law, R., Denning, A., Rayner, P., Baker, D., Bousquet, P., Bruhwiler, L., Chen, Y., Ciais, P., Fan, S., Fung, I., Gloor, M., Heimann, M., Higuchi, K., John, J., Maki, T., Maksyutov, S., Masarie, K., Peylin, P., Prather, M., Pak, B., Rander-son, J., Sarmiento, J., Taguchi, S., Takahashi, T., and Yuen, C. A.: Towards robust regional estimates of CO<sub>2</sub> sources and sinks using atmospheric transport models, *Nature*, 415, 626–630, <https://doi.org/10.1038/415626a>, 2002.
- Hobbs, J., Braverman, A., Cressie, N., Granat, R., and Gun-son, M.: Simulation-Based Uncertainty Quantification for Estimating Atmospheric CO<sub>2</sub> from Satellite Data, *SIAM/ASA Journal on Uncertainty Quantification*, 5, 956–985, <https://doi.org/10.1137/16M1060765>, 2017.
- Hobbs, J., Katzfuss, M., Zilber, D., Brynjarsdóttir, J., Mondal, A., and Berrocal, V.: Spatial Retrievals of Atmospheric Carbon Dioxide from Satellite Observations, *Remote Sensing*, 13, 571, <https://doi.org/10.3390/rs13040571>, 2021.
- Imasu, R., Matsunaga, T., Nakajima, M., Yoshida, Y., Shiomi, K., Morino, I., Saitoh, N., Niwa, Y., Someya, Y., Oishi, Y., Hashimoto, M., Noda, H., Hikosaka, K., Uchino, O., Maksyutov, S., Takagi, H., Ishida, H., Nakajima, T. Y., Nakajima, T., and Shi, C.: Greenhouse gases Observing SATellite 2 (GOSAT-2): mission overview, *Progress in Earth and Planetary Science*, 10, 33, <https://doi.org/10.1186/s40645-023-00562-2>, 2023.
- Innes, M.: Don't Unroll Adjoint: Differentiating SSA-Form Programs, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.1810.07951>, 9 March 2019.
- IPCC: Summary for Policymakers, IPCC, 1–34, <https://doi.org/10.1017/CBO9781107415324.004>, 2023.
- Johnson, S. G.: The Sobol module for Julia, GitHub [code], <https://github.com/JuliaMath/Sobol.jl> (last access: 31 January 2025), 2020.
- Kaipio, J. and Somersalo, E.: *Statistical and Computational Inverse Problems*, Springer, <https://doi.org/10.1007/b138659>, 2005.
- Kalnay, E.: *Atmospheric Modeling, Data Assimilation and Predictability*, Cambridge University Press, <https://doi.org/10.1017/CBO9780511802270>, 2002.
- Kasahara, M., Kachi, M., Inaoka, K., Fujii, H., Kubota, T., Shimada, R., and Kojima, Y.: Overview and current status of GOSAT-GW mission and AMSR3 instrument, in: *Sensors, Systems, and Next-Generation Satellites XXIV*, 21–25 September 2020, edited by: Neeck, S. P., Hélière, A., and Kimura, T., International Society for Optics and Photonics, SPIE, 11530, p. 1153007, <https://doi.org/10.1117/12.2573914>, 2020.
- Kiel, M., O'Dell, C. W., Fisher, B., Eldering, A., Nassar, R., MacDonald, C. G., and Wennberg, P. O.: How bias correction goes wrong: measurement of XCO<sub>2</sub> affected by erroneous surface pressure estimates, *Atmos. Meas. Tech.*, 12, 2241–2259, <https://doi.org/10.5194/amt-12-2241-2019>, 2019.
- Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.1412.6980>, 30 January 2017.
- Kuze, A., Suto, H., Nakajima, M., and Hamazaki, T.: Thermal and near infrared sensor for carbon observation Fourier-transform spectrometer on the Greenhouse Gases Observing Satellite for greenhouse gases monitoring, *Appl. Optics*, 48, 6716–6733, <https://doi.org/10.1364/AO.48.006716>, 2009.
- Lamminpää, O.: Forward Model Emulator for Atmospheric Radiative Transfer Using Gaussian Processes And Cross Validation, OSF [code/data set], <https://doi.org/10.17605/OSF.IO/U2T8A>, 2024.
- Lamminpää, O., Hobbs, J., Brynjarsdóttir, J., Laine, M., Braverman, A., Lindqvist, H., and Tamminen, J.: Accelerated MCMC for Satellite-Based Measurements of Atmospheric CO<sub>2</sub>, *Remote Sensing*, 11, 2061, <https://doi.org/10.3390/rs11172061>, 2019.
- Li, Z., Huang, D. Z., Liu, B., and Anandkumar, A.: Fourier Neural Operator with Learned Deformations for PDEs on General Geometries, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.2207.05209>, 2 May 2024.
- Liu, J., Bowman, K. W., Schimel, D. S., Parazoo, N. C., Jiang, Z., Lee, M., Bloom, A. A., Wunch, D., Frankenberg, C., Sun, Y., O'Dell, C. W., Gurney, K. R., Menemenlis, D., Gierach, M., Crisp, D., and Eldering, A.: Contrasting carbon cycle responses of the tropical continents to the 2015–2016 El Niño, *Science*, 358, eaam5690, <https://doi.org/10.1126/science.aam5690>, 2017.
- Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E.: Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, *Nature Machine Intelligence*, 3, 218–229, <https://doi.org/10.1038/s42256-021-00302-5>, 2021.
- Ma, P., Mondal, A., Konomi, B. A., Hobbs, J., Song, J. J., and Kang, E. L.: Computer Model Emulation with High-Dimensional Functional Output in Large-Scale Observing System Uncertainty Experiments, *Technometrics*, 64, 65–79, <https://doi.org/10.1080/00401706.2021.1895890>, 2019.
- McDuffie, J., Bowman, K., Hobbs, J., Natraj, V., Sarkissian, E., Mike, M. T., and Val, S.: Reusable Framework for Retrieval of Atmospheric Composition (ReFRACtor), Version 1.09, Zenodo [code], <https://doi.org/10.5281/zenodo.4019567>, 2020.
- Mishra, S. and Molinaro, R.: Physics informed neural networks for simulating radiative transfer, *J. Quant. Spectrosc. Ra.*, 270, 107705, <https://doi.org/10.1016/J.JQSRT.2021.107705>, 2021.
- Moore III, B., Crowell, S. M. R., Rayner, P. J., Kumer, J., O'Dell, C. W., O'Brien, D., Utembe, S., Polonsky, I., Schimel, D., and Lemen, J.: The Potential of the Geostationary Carbon Cycle Observatory (GeoCarb) to Provide Multi-scale Constraints on the Carbon Cycle in the Americas, *Front. Environ. Sci.*, 6, 109, <https://doi.org/10.3389/fenvs.2018.00109>, 2018.
- Nguyen, H. and Hobbs, J.: Intercomparison of Remote Sensing Retrievals: An Examination of Prior-Induced Biases in Averaging Kernel Corrections, *Remote Sensing*, 12, 3239, <https://doi.org/10.3390/rs12193239>, 2020.
- O'Dell, C. W., Connor, B., Bösch, H., O'Brien, D., Frankenberg, C., Castano, R., Christi, M., Eldering, D., Fisher, B., Gunson, M., McDuffie, J., Miller, C. E., Natraj, V., Oyafuso, F., Polonsky, I., Smyth, M., Taylor, T., Toon, G. C., Wennberg, P. O., and Wunch, D.: The ACOS CO<sub>2</sub> retrieval algorithm – Part I: Description and validation against synthetic observations, *Atmos. Meas. Tech.*, 5, 99–121, <https://doi.org/10.5194/amt-5-99-2012>, 2012.
- O'Dell, C. W., Eldering, A., Wennberg, P. O., Crisp, D., Gunson, M. R., Fisher, B., Frankenberg, C., Kiel, M., Lindqvist, H., Mandrake, L., Merrelli, A., Natraj, V., Nelson, R. R., Osterman, G. B., Payne, V. H., Taylor, T. E., Wunch, D., Drouin, B. J., Oyafuso, F., Chang, A., McDuffie, J., Smyth, M., Baker, D. F., Basu, S., Chevallier, F., Crowell, S. M. R., Feng, L., Palmer, P. I., Dubey,

- M., García, O. E., Griffith, D. W. T., Hase, F., Iraci, L. T., Kivi, R., Morino, I., Notholt, J., Ohyama, H., Petri, C., Roehl, C. M., Sha, M. K., Strong, K., Sussmann, R., Te, Y., Uchino, O., and Velazco, V. A.: Improved retrievals of carbon dioxide from Orbiting Carbon Observatory-2 with the version 8 ACOS algorithm, *Atmos. Meas. Tech.*, 11, 6539–6576, <https://doi.org/10.5194/amt-11-6539-2018>, 2018.
- Owhadi, H. and Yoo, G. R.: Kernel Flows: From learning kernels from data into the abyss, *J. Comput. Phys.*, 389, 22–47, <https://doi.org/10.1016/j.jcp.2019.03.040>, 2019.
- Palmer, P. I., Feng, L., Baker, D., Chevallier, F., Bösch, H., and Somkuti, P.: Net carbon emissions from African biosphere dominate pan-tropical atmospheric CO<sub>2</sub> signal, *Nat. Commun.*, 10, 3344, <https://doi.org/10.1038/s41467-019-11097-w>, 2019.
- Patil, P., Kuusela, M., and Hobbs, J.: Objective Frequentist Uncertainty Quantification for Atmospheric CO<sub>2</sub> Retrievals, *SIAM/ASA Journal on Uncertainty Quantification*, 10, 827–859, <https://doi.org/10.1137/20M1356403>, 2022.
- Patra, P., Crisp, D., Kaiser, J., Wunch, D., Saeki, T., Ichii, K., Sekiya, T., Wennberg, P., Feist, D., Pollard, D., Griffith, D., Velazco, V., Maziere, M., Sha, M., Roehl, C., Chatterjee, A., and Ishijima, K.: The Orbiting Carbon Observatory (OCO-2) tracks 2–3 peta-gram increase in carbon release to the atmosphere during the 2014–2016 El Niño, *Sci. Rep.*, 7, 13567, <https://doi.org/10.1038/s41598-017-13459-0>, 2007.
- Peiro, H., Crowell, S., Schuh, A., Baker, D. F., O'Dell, C., Jacobson, A. R., Chevallier, F., Liu, J., Eldering, A., Crisp, D., Deng, F., Weir, B., Basu, S., Johnson, M. S., Philip, S., and Baker, I.: Four years of global carbon cycle observed from the Orbiting Carbon Observatory 2 (OCO-2) version 9 and in situ data and comparison to OCO-2 version 7, *Atmos. Chem. Phys.*, 22, 1097–1130, <https://doi.org/10.5194/acp-22-1097-2022>, 2022.
- Peters, W., Jacobson, A. R., Sweeney, C., Andrews, A. E., Conway, T. J., Masarie, K., Miller, J. B., Bruhwiler, L. M. P., Pétron, G., Hirsch, A. I., Worthy, D. E. J., van der Werf, G. R., Randerston, J. T., Wennberg, P. O., Krol, M. C., and Tans, P. P.: An Atmospheric Perspective on North American Carbon Dioxide Exchange: CarbonTracker, *P. Natl. Acad. Sci. USA*, 104, 18925–18930, <https://doi.org/10.1073/pnas.0708986104>, 2007.
- Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T.: *Numerical Recipes in FORTRAN 77: The Art of Scientific Computing*, Cambridge University Press, 2nd edn., ISBN 052143064X, 1992.
- Ran, Y. and Li, X.: TanSat: a new star in global carbon monitoring from China, *Sci. Bull.*, 64, 284–285, <https://doi.org/10.1016/j.scib.2019.01.019>, 2019.
- Rasmussen, C. and Williams, C.: *Gaussian Processes for Machine Learning*, Adaptive Computation and Machine Learning, MIT Press, Cambridge, MA, USA, <http://gaussianprocess.org/gpml/> (last access: 31 January 2025), 2006.
- Rodgers, C. D.: *Inverse Methods for Atmospheric Sounding: Theory and Practice*, World Scientific Publishing Co. Pte. Ltd., Singapore 596224, Reprint edn., ISBN 981022740X, 2004.
- Rosenberg, R., Maxwell, S., Johnson, B. C., Chapsky, L., Lee, R. A. M., and Pollock, R.: Preflight Radiometric Calibration of Orbiting Carbon Observatory 2, *IEEE T. Geosci. Remote*, 55, 1994–2006, <https://doi.org/10.1109/TGRS.2016.2634023>, 2017.
- Schimel, D., Pavlick, R., Fisher, J. B., Asner, G. P., Saatchi, S., Townsend, P., Miller, C., Frankenberg, C., Hibbard, K., and Cox, P.: Observing terrestrial ecosystems and the carbon cycle from space, *Glob. Change Biol.*, 21, 1762–1776, <https://doi.org/10.1111/gcb.12822>, 2015.
- Sierk, B., Bézy, J.-L., Löscher, A., and Meijer, Y.: The European CO<sub>2</sub> Monitoring Mission: observing anthropogenic greenhouse gas emissions from space, in: *International Conference on Space Optics – ICSO 2018*, Chania, Greece, 9–12 October 2018, edited by: Sodnik, Z., Karafolas, N., and Cugny, B., International Society for Optics and Photonics, SPIE, 11180, p. 111800M, <https://doi.org/10.1117/12.2535941>, 2019.
- Sobol, I.: Distribution of Points in a Cube and Approximate Evaluation of Integrals, *Zh. Vych. Mat. Mat. Fiz.*, 7, 784–802, 1967.
- Stein, M. L.: *Interpolation of spatial data: some theory for kriging*, Springer Science & Business Media, <https://doi.org/10.1007/978-1-4612-1494-6>, 1999.
- Stewart, G. W.: *Matrix algorithms, Volume I: Basic Decompositions*, SIAM (Society for Industrial and Applied Mathematics), i–xix, <https://doi.org/10.1137/1.9781611971408.fm>, 1998.
- Tukiainen, S., Railo, J., Laine, M., Hakkarainen, J., Kivi, R., Heikkinen, P., Chen, H., and Tamminen, J.: Retrieval of atmospheric CH<sub>4</sub> profiles from Fourier transform infrared data using dimension reduction and MCMC, *J. Geophys. Res.-Atmos.*, 121, 10312–10327, <https://doi.org/10.1002/2015JD024657>, 2016.
- Vecchia, A. V.: *Estimation and Model Identification for Continuous Spatial Processes*, *J. Roy. Stat. Soc. B Met.*, 50, 297–312, <http://www.jstor.org/stable/2345768> (last access: 31 January 2025), 1988.
- Veefkind, J., Aben, I., McMullan, K., Förster, H., de Vries, J., Otter, G., Claas, J., Eskes, H., de Haan, J., Kleipool, Q., van Weele, M., Hasekamp, O., Hoogeveen, R., Landgraf, J., Snel, R., Tol, P., Ingmann, P., Voors, R., Kruizinga, B., Vink, R., Visser, H., and Levelt, P.: TROPOMI on the ESA Sentinel-5 Precursor: A GMES mission for global observations of the atmospheric composition for climate, air quality and ozone layer applications, *Remote Sens. Environ.*, 120, 70–83, <https://doi.org/10.1016/j.rse.2011.09.027>, 2012.
- Wu, K., Yang, D., Liu, Y., Cai, Z., Zhou, M., Feng, L., and Palmer, P. I.: Evaluating the Ability of the Pre-Launch TanSat-2 Satellite to Quantify Urban CO<sub>2</sub> Emissions, *Remote Sensing*, 15, 4904, <https://doi.org/10.3390/rs15204904>, 2023.
- Wunch, D., Wennberg, P. O., Osterman, G., Fisher, B., Naylor, B., Roehl, C. M., O'Dell, C., Mandrake, L., Viatte, C., Kiel, M., Griffith, D. W. T., Deutscher, N. M., Velazco, V. A., Notholt, J., Warneke, T., Petri, C., De Maziere, M., Sha, M. K., Sussmann, R., Rettinger, M., Pollard, D., Robinson, J., Morino, I., Uchino, O., Hase, F., Blumenstock, T., Feist, D. G., Arnold, S. G., Strong, K., Mendonca, J., Kivi, R., Heikkinen, P., Iraci, L., Podolske, J., Hillyard, P. W., Kawakami, S., Dubey, M. K., Parker, H. A., Sepulveda, E., García, O. E., Te, Y., Jeseck, P., Gunson, M. R., Crisp, D., and Eldering, A.: Comparisons of the Orbiting Carbon Observatory-2 (OCO-2) XCO<sub>2</sub> measurements with TCCON, *Atmos. Meas. Tech.*, 10, 2209–2238, <https://doi.org/10.5194/amt-10-2209-2017>, 2017.
- Zhu, X., Huang, L., Ibrahim, C., Lee, E. H., and Bindel, D.: Scalable Bayesian Transformed Gaussian Processes, *arXiv [preprint]*, <https://doi.org/10.48550/arXiv.2210.10973>, 20 October 2022.