Geosci. Model Dev., 13, 3439–3463, 2020
https://doi.org/10.5194/gmd-13-3439-2020

Geoscientific Model Development
Open Access
EGU

Efficient multi-scale Gaussian process regression for massive remote sensing data with satGP v0.1.2

Jouni Susiluoto[1,2,3,4], **Alessio Spantini**[1], **Heikki Haario**[2,3], **Teemu Härkönen**[2], **and Youssef Marzouk**[1]

[1]Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue, 33-207, Cambridge, MA 02139, USA
[2]Lappeenranta University of Technology, School of Engineering Science, P.O. Box 20, 53851 Lappeenranta, Finland
[3]Finnish Meteorological Institute, Erik Palménin aukio 1, 00560 Helsinki, Finland
[4]Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA

Correspondence: Jouni Susiluoto (jsusiluo@mit.edu)

Received: 28 May 2019 – Discussion started: 30 August 2019
Revised: 13 June 2020 – Accepted: 18 June 2020 – Published: 31 July 2020

Abstract. Satellite remote sensing provides a global view to processes on Earth that has unique benefits compared to making measurements on the ground, such as global coverage and enormous data volume. The typical downsides are spatial and temporal gaps and potentially low data quality. Meaningful statistical inference from such data requires overcoming these problems and developing efficient and robust computational tools. We design and implement a computationally efficient multi-scale Gaussian process (GP) software package, satGP, geared towards remote sensing applications. The software is able to handle problems of enormous sizes and to compute marginals and sample from the random field conditioning on at least hundreds of millions of observations. This is achieved by optimizing the computation by, e.g., randomization and splitting the problem into parallel local subproblems which aggressively discard uninformative data.

We describe the mean function of the Gaussian process by approximating marginals of a Markov random field (MRF). Variability around the mean is modeled with a multi-scale covariance kernel, which consists of Matérn, exponential, and periodic components. We also demonstrate how winds can be used to inform covariances locally. The covariance kernel parameters are learned by calculating an approximate marginal maximum likelihood estimate, and the validity of both the multi-scale approach and the method used to learn the kernel parameters is verified in synthetic experiments.

We apply these techniques to a moderate size ozone data set produced by an atmospheric chemistry model and to the very large number of observations retrieved from the Orbiting Carbon Observatory 2 (OCO-2) satellite. The satGP software is released under an open-source license.

1 Introduction

Climate change is one of the most important present-day global environmental challenges. The underlying reason is anthropogenic carbon emissions. According to the Intergovernmental Panel on Climate Change, carbon dioxide (CO_2) has the strongest effect on warming the planet of the well-mixed greenhouse gases, with the radiative forcing of ca. 1.68 W m^{-2} (IPCC, 2013).

Several instruments orbiting the Earth produce enormous quantities of remote sensing data, used to compute local estimates of CO_2 and other atmospheric constituents by solving complicated inverse problems and further processed to, e.g., gridded data products and flux estimates (Cressie, 2018). These instruments include the Greenhouse gases Observing SATellite (GOSAT) from Japan (Yokota et al., 2009), operational since January 2009; the OCO-2 from NASA (Crisp et al., 2012), launched in July 2014; and the Chinese TanSat (Yi et al., 2018), launched in December 2016. GOSAT-2 was launched in October 2018, and in May 2019 the OCO-3 instrument (Eldering et al., 2019) was taken to the International Space Station. In addition to the CO_2-measuring instruments, also other types of data are produced by remote sensing. For instance the European TROPOspheric Monitoring Instrument (TROPOMI)

produces measurements of nitrogen dioxide, formaldehyde, carbon monoxide, aerosols, methane, and ozone. Common denominators among most non-gridded remote sensing data sets include a large number of observations, global coverage but small area observed at any given time, sensitivity to prevailing weather conditions and cloud cover, unknown and/or unreported error covariances, and predetermined positioning that rules out freely observing at a given time and location. These shortcomings can be partly remedied with techniques from computational statistics, such as those implemented in the satGP software, which this paper introduces.

There are two key advances in this work. First, we describe the computational approaches that allow satGP to tackle remote-sensing-related spatial statistics problems of enormous sizes. Second, we present formulations of a multi-scale covariance function and a space-dependent mean function, types of which we have not seen used in the remote sensing community. We also show how these functions can be efficiently learned from data.

Related to this work, several kriging studies have been published before in the context of remotely sensed CO_2. Zeng et al. (2013) analyzed the variability in CO_2 in both space and time over China and produced monthly maps from GOSAT data with slightly over 10 000 observations. Nguyen et al. (2014) used a 4-times-larger set of observations with Kalman smoothing in a reduced dimension with GOSAT and the Atmospheric InfraRed Sounder (AIRS) data from NASA. A map of atmospheric carbon dioxide derived from GOSAT data was presented at the higher resolution of $1° \times 1.25°$ in space and 6 d in time by Hammerling et al. (2012). In another publication by the same authors, synthetic OCO-2 observations were considered with the same spatial resolution.

More recently Zeng et al. (2017) presented a global data set derived from GOSAT with the spatiotemporal resolution of 3 d and 1°, and this study evaluated also the temporal trend of the XCO_2. The results were validated against observations from the Total Carbon Column Observing Network (TCCON) and against modeling results from CarbonTracker and Goddard Earth Observing System with atmospheric chemistry (GEOS-Chem). Tadić et al. (2017) described a moving-window block kriging algorithm to introduce time dependence into a GOSAT-based XCO_2 map construction process using a quasi-probabilistic screening method for subsampling observations, thinning the data for computational reasons. Other recent studies have also contained analyses of OCO-2 data – for example Zammit-Mangion et al. (2018) presented fixed rank kriging results based on OCO-2 data using a 16 d moving window. In many of these studies, the obtained CO_2 fields appear very smooth.

Applications to remote sensing data have also resulted in publications more focused on methods. Ma and Kang (2020) described a "fused" Gaussian process, combining a graphical model with a Gaussian process and applying that to sea surface temperature data. In another computationally sophisticated application, Zammit-Mangion et al. (2015) simultaneously modeled both flux fields and concentrations using a bivariate spatiotemporal model with Hamiltonian Monte Carlo (Neal, 2011) for sampling the posterior. Due to computational challenges the spatial area investigated in this work was very small.

For Gaussian processes, various approaches have been studied to overcome the difficulties posed by large amounts of data. For instance, Lindgren et al. (2011) provide an explicit link between some random fields arising as solutions to certain stochastic partial differential equations and Markov random fields. A recent review of Vecchia-type approximations (Vecchia, 1988) is given by Katzfuss et al. (2018), and Heaton et al. (2018) presents a comparison of the performance of several recently developed spatial statistics methods with applications to data from the Moderate Resolution Imaging Spectroradiometer (MODIS). The difficulty of ordering the observations for effective inference with Gaussian processes, especially as the dimension of the inputs grows, is discussed by Ambikasaran et al. (2016).

In this work we describe the satGP program, which solves very large spatiotemporal statistics problems with up to at least the order of 10^8 marginals conditioned on 10^8 observations. While advances have recently been made in the field, we are not aware of any literature or software solving problems of quite this scale so far. The effectiveness is partly based on combining ideas related to Vecchia-type and nearest-neighbor Gaussian processes (Datta et al., 2016), but satGP also employs several computational tricks such as subsampling observations and filtering out uninformative data at several levels when possible. The program includes a flexible implementation for space-dependent mean functions and space-independent covariance kernels and routines for learning their parameters from data. The spatial dependence of the mean function is learned by computing marginals of a Markov random field (MRF). The covariance function is constructed in a way that allows for describing the multiple natural length scales in the data. After learning the model parameters the program computes posterior predictive fields, and realizations can be drawn from both the posterior and the prior.

We validate the multi-scale covariance modeling approach by learning the covariance function parameters of a data set drawn with satGP from the prior of a multi-scale Gaussian process. To demonstrate the computational capabilities of this early version of satGP, we computed global XCO_2 concentrations for a duration of 1526 d at 0.5° spatial and daily temporal resolution, amounting to calculating 350 million marginal distributions, conditioning on 116 million XCO_2 observations from OCO-2. Figure 9 shows an example of what these results look like. We also present a nonstationary covariance kernel formulation that utilizes wind data for computation, and we use that covariance function with OCO-2 data. The utility of using winds with CO_2 data has been demonstrated before by, e.g., Nassar et al. (2017).

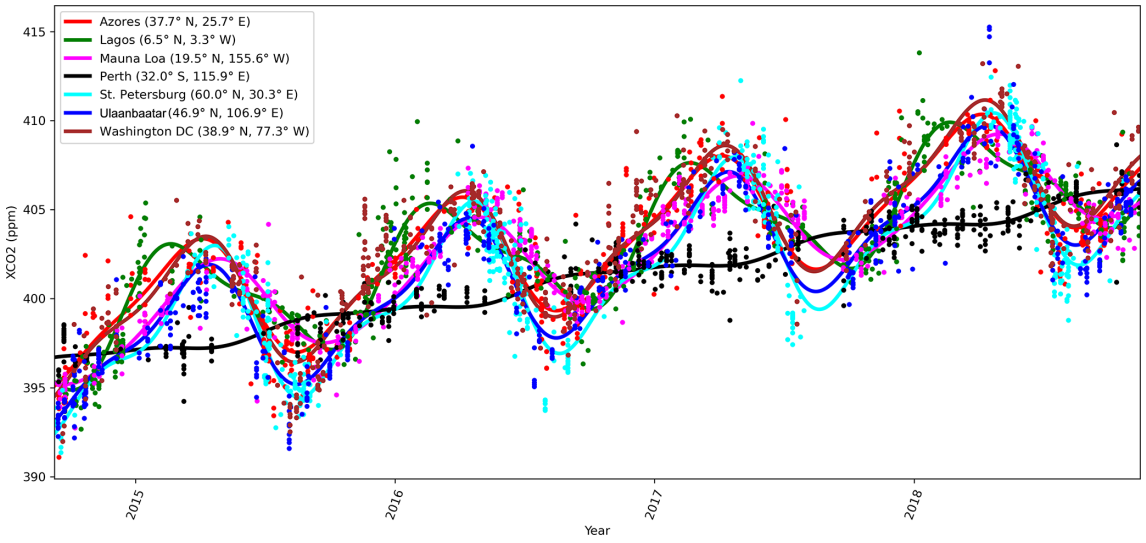

Figure 1. Mean function m with components f_i given by Eq. (11). The solid lines show the mean function value for each day, fitted to the XCO_2 observations, marked by the dots. The OCO-2 mean function results are discussed in Sect. 4.4.

In addition to the OCO-2 work, we demonstrate the capabilities of satGP with synthetic ozone data from the Whole Atmosphere Community Climate Model (WACCM4) (Marsh et al., 2013), emulating observing with the Global Ozone Monitoring by Occultation of Stars (GOMOS) instrument (Bertaux et al., 2004, 2010; Kyrölä et al., 2004) on Envisat. Using synthetic data allows us to directly compare Gaussian process posterior estimates to an exactly known ground truth. The software could equally well be applied to any other observed quantity of interest.

The rest of the paper is organized in the following manner. Section 2 describes the methods both generally and as implemented in satGP. Section 3 discusses the computational details in satGP. Section 4 presents and discusses simulation results, including a multi-scale synthetic parameter identifiability study, an application to synthetic WACCM4-generated data, and applications using the OCO-2 V9 data. In the concluding Sect. 5, current limitations and some possible future directions are briefly mentioned.

2 Methods

In geosciences, kriging (Cressie and Wikle, 2001; Chiles and Delfiner, 2012) is used for performing spatial statistics tasks such as gap-filling or representing data in a grid. The semivariogram models used in kriging are closely related to the covariance models used in the Gaussian process formalism (Santner et al., 2003; Rasmussen and Williams, 2006; Gelman et al., 2013), where instead of learning the variogram model from the data, a form of a covariance function is prescribed and its parameters estimated.

With Gaussian processes, we want to learn properties of a spatiotemporal surface from some observational data of some quantity of interest. To each point in space and time corresponds a Gaussian distribution of that quantity, whose mean and variance can be calculated by solving a local regression problem. This is closely related to solving a spatiotemporal interpolation problem when the observations have Gaussian errors.

The theory of Bayesian statistics, Gaussian processes, and Markov random fields that is used in this work is well known, and therefore, many of the novel aspects in this section have to do with the computational methods and modifications that are presented, such as observation selection schemes in Sect. 2.5 or approximate marginal maximum likelihood computation in Sect. 2.6. These modifications trade precision for tractability but in a way that tries minimize the loss in accuracy. Due to the desire to be able to solve very large problems, some sacrifices need to be made to be able to obtain any solution.

This section goes through the Gaussian process formalism and presents both generic and satGP-specific forms of mean and covariance functions. This is followed by discussion of how observation selection is carried out for solving local subproblems and how model parameters are learned. The presentation of the general Gaussian process problem is based on Santner et al. (2003) and Rasmussen and Williams (2006). Commonly used notation is listed in Table 1.

2.1 Gaussian process regression

A Gaussian process is a stochastic process, which can be thought of as an infinite-dimensional Gaussian distribution in that the joint distributions of the process at any finite set of space–time points are multivariate normal. We denote points in the spatiotemporal domain by $\boldsymbol{x} \in \mathbb{R}^q$. In this work $q = 3$,

Table 1. Most commonly used notation related to inputs and mean/covariance functions in Sect. 2 and the Markov random field in Sect. 2.3.1. The second column gives the set in which the symbol belongs – or in some cases the set of which the symbol is a subset. The domain sets in the second column are defined as $D_{\mathrm{lat}} \triangleq \left[\mathrm{lat}_{\min}, \mathrm{lat}_{\max}\right]$, $D_{\mathrm{lon}} \triangleq \left[\mathrm{lon}_{\min}, \mathrm{lon}_{\max}\right]$, $D_{\mathrm{t}} \triangleq \mathbb{R}^+$, and $D \triangleq D_{\mathrm{lat}} \times D_{\mathrm{lon}} \times D_{\mathrm{t}} \subset \mathbb{R}^q$, and $\mathcal{V}$ denotes the set of nodes in the graph described in Sect. 2.3.1.

Symbol	$\in$	Meaning
$\boldsymbol{x}$	D	Generic spatiotemporal coordinate vector
x^{t}	D_{t}	Temporal part of coordinate vector $\boldsymbol{x}$, implemented as seconds since 1970
$\boldsymbol{x}^{\mathrm{s}}$	$\mathbb{R}^{q-1}$	Spatial part of generic coordinate $\boldsymbol{x}$, in practice $\boldsymbol{x}^{\mathrm{s}} = [x^{\mathrm{lat}}, x^{\mathrm{lon}}]^T$
x^{lat}	D_{lat}	North–south component of coordinate vector $\boldsymbol{x}$ as defined by variable area in Table 2
x^{lon}	D_{lon}	East–west component of coordinate vector $\boldsymbol{x}$ as defined by variable area in Table 2
$\boldsymbol{x}^{ij}$	$D_{\mathrm{lat}} \times D_{\mathrm{lon}}$	Spatial location corresponding to ith latitude and jth longitude in the satGP regular grid
$\boldsymbol{x}^*$	$\mathbb{R}^q$	Gaussian process test input – the spatiotemporal location where the GP is evaluated
$\mathbf{x}^{\mathrm{obs}}$	$\mathbb{R}^{n \times q}$	Matrix of space–time locations where the n observations in $\boldsymbol{\psi}^{\mathrm{obs}}$ were made
$\boldsymbol{\beta}$	$\mathbb{R}^{n_\beta}$	Mean function coefficients; see m below. These may be space dependent.
$\boldsymbol{\beta}_\nu$	$\mathbb{R}^{n_\beta}$	$\boldsymbol{\beta}$ coefficients for the spatial location corresponding to graph label ν in the MRF
$\boldsymbol{\beta}^{ij}$	$\mathbb{R}^{n_\beta}$	$\boldsymbol{\beta}$ coefficients at grid point $\boldsymbol{x}^{ij}$ in the satGP latitude–longitude grid
$\boldsymbol{\beta}_{\mathcal{V}}$	$\mathbb{R}^{n_\beta \times \mathcal{V}}$	$\boldsymbol{\beta}$ coefficients for all grid points in the satGP latitude–longitude grid
$\boldsymbol{\delta}$	$\mathbb{R}^{n_\delta}$	Space-dependent mean function parameters that cannot be learned via Eqs. (6) and (7)
$\boldsymbol{\delta}_\nu$	$\mathbb{R}^{n_\delta}$	$\boldsymbol{\delta}$ parameters for the spatial location corresponding to graph label ν in the MRF
$\boldsymbol{\delta}_{\mathcal{V}}$	$\mathbb{R}^{n_\delta \times \mathcal{V}}$	$\boldsymbol{\delta}$ coefficients for all grid points in the satGP latitude–longitude grid
$\boldsymbol{\theta}$	$\mathbb{R}^{n_\theta}$	Covariance function parameters of all the subkernels of the multi-scale kernel
$\boldsymbol{\theta}_{(\cdot)}$	$\mathbb{R}^{n_{\theta_{(\cdot)}}}$	Covariance function parameters of the subkernel in the subindex $(\cdot)$
I	–	The set of all spatial/temporal indexes for each $\boldsymbol{x}$; size of $\lvert I \rvert$ is therefore q.
I_{ST}	$\subseteq I$	Spatiotemporal index set: corresponding k is a function of space and time.
I_{S}	$\subseteq I$	Spatial index set: corresponding k is a function of space only.
$\ell_c, c \in I$	$\mathbb{R}^+$	Covariance kernel length-scale parameter along axis c
$\boldsymbol{\ell}_{I'}$	$\mathbb{R}^{+\lvert I' \rvert}$	Covariance kernel length-scale parameters along all dimensions in I'
ν	$\mathcal{V}$	Label of a specific node of the graph describing the MRF. In Sect. 2.4 ν is a parameter ($\in \mathbb{R}^+$) used to define the Matérn kernel smoothness parameter.
ν^{ij}	$\mathcal{V}$	Label of the node of the graph corresponding to the spatial location of $\boldsymbol{x}^{ij}$
∂_ν	$\subseteq \mathcal{V}$	Set of nodes in the graph with edges to node ν
Ψ	–	Random field of the quantity of interest
$\Psi(\boldsymbol{x})$	$\mathbb{R}$	Random variable of the quantity of interest corresponding to Ψ at $\boldsymbol{x}$
$\boldsymbol{\psi}^{\mathrm{obs}}$	$\mathbb{R}^n$	Values of the observations of the field at locations $\mathbf{x}^{\mathrm{obs}}$
$\boldsymbol{\psi}$	$\mathbb{R}^D$	Realization of the random field Ψ
$k(\boldsymbol{x}, \boldsymbol{x}')$	$\mathbb{R}$	Covariance function value of inputs $\boldsymbol{x}$ and $\boldsymbol{x}'$
$m(\boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\delta})$	$\mathbb{R}$	Mean function value at $\boldsymbol{x}$ with parameters $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$: $m(\boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\delta}) = f(\boldsymbol{x}; \boldsymbol{\delta})^T \boldsymbol{\beta}$
$\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\delta})$	$\mathbb{R}$	Vector of functions to construct the mean function at $\boldsymbol{x}$ with parameters $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$
$\mathbf{F}$	$\mathbb{R}^{n \times n_\beta}$	Matrix with coefficients $\mathbf{F}_{ij} = f_j(\mathbf{x}_i^{\mathrm{obs}}; \boldsymbol{\delta})$
$\mathbf{K}$	$\mathbb{R}^{n \times n}$	Covariance matrix with elements $\mathbf{K}_{ij} = \boldsymbol{k}(\boldsymbol{x}_i^{\mathrm{obs}}, \boldsymbol{x}_j^{\mathrm{obs}})$

even though this restriction can be overcome if needed, and satGP does have limited support for space-only problems.

The Gaussian process, or Gaussian random field, is denoted by

$$\Psi \sim \mathrm{GP}(m(\boldsymbol{x}; \boldsymbol{\beta}), k(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})), \tag{1}$$

where $m : \mathbb{R}^q \to \mathbb{R}$ and $\boldsymbol{k} : \mathbb{R}^{q \times q} \to \mathbb{R}$ are the mean and covariance functions of the process parameterized by hyperparameter vectors $\boldsymbol{\beta} \in \mathbb{R}^{n_\beta}$ and $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$. The infinite-dimensional (since the domain of $\boldsymbol{x}$ is typically infinite) description in Eq. (1) is reduced below to a finite-dimensional problem, in which case $k(\boldsymbol{x}, \boldsymbol{x}')$ describes an entry of the covariance matrix of the joint distribution of random variables $\Psi(\boldsymbol{x})$ over all $\boldsymbol{x}$ that one is interested in.

The function m above is called drift in kriging literature, and the expected value of the process in regions with no data will tend to the value of this mean function. It is chosen to reflect the deterministic patterns in the data, and the particular form picked to model m will also affect how the function k and parameters $\boldsymbol{\theta}$ in Eq. (1) need to be specified. With inad-

equate modeling of the mean function, the uncertainty estimates obtained with Gaussian process regression may end up being unnecessarily large. For instance linear trends, constant factors, and seasonal and other periodic fluctuations should be included in m if they are known. An example of what is used with the OCO-2 data is shown later in Eq. (11).

In what follows, the domain $\mathbb{R}^q \ni \boldsymbol{x}$ is divided into two disjoint parts, one of which, $\mathcal{X}^{\text{train}} \subset \mathbb{R}^q$, is the set of coordinates $\boldsymbol{x}_i^{\text{obs}}$, where observation data (training data) were measured, and another one, $\mathcal{X}^{\text{test}} \triangleq \mathbb{R}^q \setminus \mathcal{X}^{\text{train}}$, denotes its complement. Points in $\mathcal{X}^{\text{test}}$ are denoted by $\boldsymbol{x}^*$ and called *test inputs* as is often done in Gaussian process literature. Observations at locations $\{\mathbf{x}_i^{\text{obs}} : i = 1, \ldots, n\}$, both real and synthetic ones generated by the Gaussian process, are denoted by $\boldsymbol{\psi}_i^{\text{obs}} \in \mathbb{R}$, and the vector of all $\boldsymbol{\psi}_i^{\text{obs}}$ is written as $\boldsymbol{\psi}^{\text{obs}}$.

For the mean function m in Eq. (1), a specific form,

$$m(\boldsymbol{x}; \boldsymbol{\beta}, \boldsymbol{\delta}) = f(\boldsymbol{x}; \boldsymbol{\delta})^T \boldsymbol{\beta} \equiv \widetilde{f}(\boldsymbol{x}^{\text{t}}; \boldsymbol{\delta}(\boldsymbol{x}^{\text{s}}))^T \boldsymbol{\beta}(\boldsymbol{x}^{\text{s}}), \tag{2}$$

is used in this work. The superindexes s and t refer to the spatial and temporal parts of the generic coordinate $\boldsymbol{x}$, and $\boldsymbol{\delta}$ values are auxiliary parameters which are space dependent. The purpose of the righthand side with the function $\widetilde{f}$ is to underline that f depends on the spatial part of $\boldsymbol{x}$ only via the space-dependent $\boldsymbol{\delta}$ parameters and that the $\boldsymbol{\beta}$ parameters do not depend on x^{t}, the temporal part of $\boldsymbol{x}$. The temporal evolution of the mean function is in this particular form determined only by the function $f(\boldsymbol{x}; \boldsymbol{\delta}) \triangleq [f_1(\boldsymbol{x}; \boldsymbol{\delta}), \ldots, f_{n_\beta}(\boldsymbol{x}; \boldsymbol{\delta})]^T$, and for each f_i there is a space-dependent regression coefficient $\boldsymbol{\beta}_i$.

The parameter vectors $\boldsymbol{\delta}$ contains space-dependent parameters that affect the form of any of the f_i in a way that cannot be modeled with the $\boldsymbol{\beta}$ coefficients in the functional form of Eq. (2). The length of these space-dependent $\boldsymbol{\delta}$ vectors is n_δ. Given the parameters $\boldsymbol{\delta}$ for all the inputs in $\mathbf{x}^{\text{obs}}$ and a set of functions f_i for constructing the mean function, we define matrix $\mathbf{F} \in \mathbb{R}^{n \times n_\beta}$ with elements $\mathbf{F}_{ij} = f_i(\mathbf{x}_j^{\text{obs}}; \boldsymbol{\delta})$, where the $\boldsymbol{\delta}$ is now specific to the location $\mathbf{x}_i^{\text{obs}}$.

The definition of m above is very general and can describe in practice a large number of realistic scenarios. Nonetheless, the form of Eq. (2) imposes the strong assumption of separation of space and time in that the $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ parameters do not depend on time. The explicit form of functions f_i used to model the OCO-2 data are given below in Sect. 2.2.

The covariance function $k(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})$ controls the smoothness of the draws $\boldsymbol{\psi}$ from Ψ. It outputs the prior covariance of the random variables $\Psi(\boldsymbol{x})$ and $\Psi(\boldsymbol{x}')$ at $\boldsymbol{x}$ and $\boldsymbol{x}'$. The parameter vector $\boldsymbol{\theta}$ typically contains at least one scale parameter ℓ and a parameter τ controlling the maximum covariance, τ^2. The ℓ parameters correspond to the length scales of the random fluctuations of the realizations around the mean function, and the τ parameters describe the amplitude of that fluctuation. By defining the covariance matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ with elements $\mathbf{K}_{i,j} = k(\mathbf{x}_i^{\text{obs}}, \mathbf{x}_j^{\text{obs}}; \boldsymbol{\theta})$, the joint distribution of the field at observed locations is given by

$$\Psi^{\text{obs}} \sim \mathcal{N}(\mathbf{F}\boldsymbol{\beta}, \mathbf{K})). \tag{3}$$

Explicit forms of functions m and k are described in Sect. 2.2 and 2.4, respectively. Additional practical guidelines are given in Appendix A.

The paradigm of Bayesian statistics is standard for analyzing data and uncertainties, and it is also widely used in geosciences (Rodgers, 2000; Gelman et al., 2013). Given the observed data $\Psi^{\text{obs}} = \boldsymbol{\psi}^{\text{obs}}$ at some finite set of points $\mathbf{x}^{\text{obs}}$, the object of interest of the Bayesian inference problem in this work is the joint posterior distribution of the Gaussian process and the parameters,

$$p(\boldsymbol{\psi}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta} | \boldsymbol{\psi}^{\text{obs}}) = \frac{p(\boldsymbol{\psi}^{\text{obs}} | \boldsymbol{\psi}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta}) p(\boldsymbol{\psi} | \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta}) p(\boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta})}{p(\boldsymbol{\psi}^{\text{obs}})}, \tag{4}$$

where $p(\boldsymbol{\psi} | \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta})$ is the Gaussian process prior, and $p(\boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\theta})$ is a prior on the Gaussian process hyperparameters. In this particular equation $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ actually denote spatially varying hyperparameter fields. The calculation in Eq. (4) is not generally tractable for a huge number of inputs $\boldsymbol{x}$, but posterior estimates of the GP, $p(\boldsymbol{\psi} | \boldsymbol{\psi}^{\text{obs}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\theta}})$, can be calculated for a finite set of inputs by conditioning on parameter point estimates $\widehat{\boldsymbol{\theta}}$, $\widehat{\boldsymbol{\beta}}$, and $\widehat{\boldsymbol{\delta}}$. The covariance parameter estimate $\widehat{\boldsymbol{\theta}}$ may be found by minimizing some loss function $\mathcal{L}$,

$$\widehat{\boldsymbol{\theta}} = \arg\,\min_\theta\, \mathcal{L}(\boldsymbol{\theta}), \tag{5}$$

described explicitly below in Sect. 2.6. Given a point estimate of the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\delta}$, along with a Gaussian prior for the $\boldsymbol{\beta}$ parameters with mean $\boldsymbol{\mu}_\beta$ and covariance $\boldsymbol{\Sigma}_\beta$, the posterior distribution of the $\boldsymbol{\beta}$ parameters can be computed with,

$$\mathbb{E}\left[\boldsymbol{\beta} | \Psi^{\text{obs}} = \boldsymbol{\psi}^{\text{obs}}, \boldsymbol{\theta}, \boldsymbol{\delta}\right] = \left(\mathbf{F}^T \mathbf{K}^{-1} \mathbf{F} + \boldsymbol{\Sigma}_\beta^{-1}\right)^{-1} \mathbf{F}^T \mathbf{K}^{-1} \left(\boldsymbol{\psi}^{\text{obs}} - \mathbf{F}\boldsymbol{\mu}_\beta\right) + \boldsymbol{\mu}_\beta \tag{6}$$

$$\text{Cov}\left[\boldsymbol{\beta} | \Psi^{\text{obs}} = \boldsymbol{\psi}^{\text{obs}}, \boldsymbol{\theta}, \boldsymbol{\delta}\right] = \left(\mathbf{F}^T \mathbf{K}^{-1} \mathbf{F} + \boldsymbol{\Sigma}_\beta^{-1}\right)^{-1}. \tag{7}$$

The matrix $\mathbf{K}$ is generally a dense matrix of size $n \times n$, where n is the number of observations, and as n may be extremely large, direct inversion of this matrix is in practice impossible. However, in this work inverting the full $\mathbf{K}$ is not necessary; we want to find parameters $\boldsymbol{\beta}$ that vary locally, which is done by splitting the full problem into many smaller subproblems, solving the $\boldsymbol{\beta}$ parameters in a grid, as described in Sect. 2.3. This grid is then used to construct matrix $\mathbf{F}$ by interpolating the values of $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ obtained.

The $\boldsymbol{\delta}$ parameters are found approximately in this work by a three-step process: first a point estimate of parameters $\boldsymbol{\beta}$

and $\boldsymbol{\delta}$ is computed using an optimization algorithm; second, parameters $\boldsymbol{\beta}$ are recomputed by Eq. (6) given the estimate of $\boldsymbol{\delta}$ from the first stage; and third, the $\boldsymbol{\delta}$ parameters alone are recalibrated by optimization using the newly found $\boldsymbol{\beta}$ parameters. In practice this procedure produces stable results with the OCO-2 data, and for pathological data sets repeated alternating optimization of the parameters may be performed. The calibration process is described in more detail in Sect. 2.3.2.

Even though a full posterior distribution of the parameters is not obtained this way, the solution of the Gaussian process itself is Bayesian in that the posterior marginals at each $\boldsymbol{x}^*$ are found by conditioning on the observations. In the satGP software, the space-dependent $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ parameters are fitted first, and any learning of the covariance parameters is done only after that.

For prediction in the context of Gaussian random functions, the properties of multivariate normal distributions are exploited for calculating marginals of the random field Ψ at any set of inputs. The posterior distribution $p(\boldsymbol{\psi}^*|\boldsymbol{\psi}^{\text{obs}}, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\beta}})$ of the Gaussian process at some test input $\boldsymbol{x}^*$ can, given point estimates $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\theta}}$, be modeled according to Eq. (3) with

$$\begin{pmatrix} \Psi^* \\ \Psi^{\text{obs}} \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} f(\boldsymbol{x}^*)^T \\ \mathbf{F} \end{bmatrix}\widehat{\boldsymbol{\beta}}, \begin{bmatrix} \mathbf{K}(\boldsymbol{x}^*, \boldsymbol{x}^*) & \mathbf{K}(\boldsymbol{x}^*, \mathbf{x}^{\text{obs}}) \\ \mathbf{K}(\mathbf{x}^{\text{obs}}, \boldsymbol{x}^*) & \mathbf{K}(\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{obs}}) \end{bmatrix}\right), \quad (8)$$

where the vector of inputs has been divided into two parts – one for the test input $\boldsymbol{x}^*$ and the other one for the observations $\mathbf{x}^{\text{obs}}$. The notation $\mathbf{K}(\boldsymbol{x}^*, \mathbf{x}^{\text{obs}})$ refers to the first row (minus the first element) of the covariance matrix with elements $\mathbf{K}(\boldsymbol{x}^*, \mathbf{x}^{\text{obs}})_j = k(\boldsymbol{x}^*, \mathbf{x}_j^{\text{obs}})$, and the matrix in the lower-right corner, $\mathbf{K}(\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{obs}})$ is the same as matrix $\mathbf{K}$ in, e.g., Eq. (3). The random variable at $\boldsymbol{x}^*$ can then be written as $\Psi^*|\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\theta}} \sim \mathcal{N}(\mu^*, \Sigma^*)$, where its mean and covariance are given by

$$\mu^* = f(\boldsymbol{x}^*)^T\widehat{\boldsymbol{\beta}} + \mathbf{K}\left(\boldsymbol{x}^*, \mathbf{x}^{\text{obs}}\right)\mathbf{K}\left(\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{obs}}\right)^{-1}\left(\boldsymbol{\psi}^{\text{obs}} - \mathbf{F}\widehat{\boldsymbol{\beta}}\right) \quad (9)$$

and

$$\Sigma^* = \mathbf{K}\left(\boldsymbol{x}^*, \boldsymbol{x}^*\right) - \mathbf{K}\left(\boldsymbol{x}^*, \mathbf{x}^{\text{obs}}\right)\mathbf{K}\left(\mathbf{x}^{\text{obs}}, \mathbf{x}^{\text{obs}}\right)^{-1}\mathbf{K}\left(\mathbf{x}^{\text{obs}}, \boldsymbol{x}^*\right), \quad (10)$$

and where the covariance Σ^* is the Schur complement of $\mathbf{K}(\boldsymbol{x}^*, \boldsymbol{x}^*)$. The formulas in Eqs. (8)–(10) work equally well when the $\boldsymbol{x}^*$ contains more than one test input. However, as of now, in satGP these equations are solved for a single test input at a time. When computing Ψ^* with these formulas, satGP uses observations close to $\boldsymbol{x}^*$ (see Sect. 2.5) and the values of $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ calibrated at $\boldsymbol{x}^{*\text{s}}$.

2.2 Mean functions in satGP

Equation (2) gives the most general mean function form available in satGP. The functions f_i above are user defined, and, for ease of use, satGP includes functionality for using a zero-mean function, a spatially independent mean function, and an arbitrary gridded array of values. The specific forms of f_i used for the OCO-2 experiments in Sect. 4 are

$$\left.\begin{aligned} f_1(\boldsymbol{x}) &= \sin\left(2\pi x^{\text{t}}\Delta_{\text{period}}^{-1} + \boldsymbol{\delta}\right) \\ f_2(\boldsymbol{x}) &= \cos\left(4\pi x^{\text{t}}\Delta_{\text{period}}^{-1} + \boldsymbol{\delta}\right) \\ f_3(\boldsymbol{x}) &= 1 \\ f_4(\boldsymbol{x}) &= x^{\text{t}}, \end{aligned}\right\} \quad (11)$$

where Δ_{period} is for OCO-2 the duration of 1 year, and $\boldsymbol{\delta}$ is a space-dependent phase shift. The function f_1 fits the summer–winter cycle, and f_2 fits the semiannual cycle. It is assumed that for any given $\boldsymbol{x}$, f_1 and f_2 can be modeled with the same $\boldsymbol{\delta}$ parameters. The constant term is given by f_3, and f_4 gives the slow global trend. As an example of the local behavior, Fig. 1 shows the mean function fit to the observed local daily mean values of XCO_2 from OCO-2 for several locations. The WACCM4 ozone study in Sect. 4.2 added two more functions f_5 and f_6 similar to f_1 and f_2 but with different Δ_{period} parameters.

2.3 Learning the spatial dependence of $\boldsymbol{\beta}$

When satGP is not used for learning GP covariance parameters or generating synthetic training sets, the finite set of test inputs $\boldsymbol{x}^*$ for GP calculation is a grid with predefined geographical and temporal extents and resolution. Solving the GP marginalization and sampling problems then amounts to solving Eqs. (9) and (10) at each corresponding space–time point. Since, e.g., sources, sinks, and timing of seasons are local, the mean function should be different from one spatial grid point to another. This is achieved by modeling the $\boldsymbol{\beta}$ parameters as a Markov random field. The MRF imposes the condition that neighboring grid cells should not be too different from each other. How different they are allowed to be is a modeling choice; see Appendix A. This section describes how the spatial dependence is resolved in satGP using computational statistics.

In addition to solving this spatial problem, the marginal distributions of the $\boldsymbol{\beta}$ parameters need to be solved for each individual vertex. Point estimates of the $\boldsymbol{\delta}$ parameters, mentioned in Sect. 2.1, are found at the same time with the $\boldsymbol{\beta}$ parameters. The intimately connected spatial and local problems are described in the subsections below.

2.3.1 Mean function parameters $\boldsymbol{\beta}$ are described as a Markov random field

A Markov random field is a probabilistic model that describes the conditional independence structure in a set of ran-

dom variables. In satGP, an MRF is used to describe how the $\boldsymbol{\beta}$ coefficients depend on each other spatially. The MRF used in satGP assumes that, in addition to data, the $\boldsymbol{\beta}$ coefficients only depend on the coefficient values in the neighboring grid points.

Technically, the MRF in satGP is an undirected graphical model $\mathcal{G} \triangleq (\mathcal{V}, \mathcal{E})$ (Lauritzen, 1996), with the set of vertices or nodes $\mathcal{V} \triangleq \{\nu^{ij} | i = 1 \ldots n_{\text{lat}}, j = 1 \ldots n_{\text{lon}}\}$ and edges $\mathcal{E} \triangleq \{(\nu^{i,j}, \nu^{i+1,j}) | i = 1 \ldots n_{\text{lat}} - 1, j = 1 \ldots n_{\text{lon}}\} \cup \{(\nu^{k,l}, \nu^{k,l+1}) | k = 1 \ldots n_{\text{lat}}, l = 1 \ldots n_{\text{lon}} - 1\}$. We use both ν and ν^{ij} to denote a generic vertex in a graph, and in the specific MRF setting used in satGP, each ν^{ij} corresponds to the random vector $\boldsymbol{\beta}^{ij}$ at grid point (i, j). After finding the marginal distributions of these vectors in the graph the maximum a posteriori (MAP) values of $\boldsymbol{\beta}^{ij}$ are used as the parameters of the mean function for the spatial location corresponding to the (i, j) element.

The set of edges $\mathcal{E}$ defines the Markov structure of the graph, i.e., how the $\boldsymbol{\beta}$ coefficients of the nodes depend on each other. For any non-edge vertex $\nu^{i,j}$, there are edges in $\mathcal{E}$ to the east, south, west, and north, meaning that only these neighboring vertices, collectively denoted by $\partial_{\nu^{ij}} \triangleq \{\nu \in \mathcal{V} | (\nu, \nu^{ij}) \in \mathcal{E}\}$, directly affect the vertex. More specifically, the Markov property defined by the set $\mathcal{E}$ implies that the probability of the $\boldsymbol{\beta}$ parameters of latitude i and longitude j is given by $p(\boldsymbol{\beta}^{ij}) = \int_{\partial\nu^{ij}} p(\nu^{ij} | \partial_{\nu^{ij}}) p(\partial_{\nu^{ij}})$, where it is understood that ν^{ij} and $\partial_{\nu^{ij}}$ refer directly to the random variables, $\boldsymbol{\beta}^{ij}$ and the joint distribution of the $\boldsymbol{\beta}$ coefficients of its adjacent vertices, respectively.

The satGP program needs to compute the marginal distributions of each $\boldsymbol{\beta}^{ij}$ to learn the spatially varying mean function parameters. Due to the lattice structure of the graph, according to Hammersley and Clifford (1971) the full joint distribution of the graph $p(\mathcal{V})$ factors as $\prod_{(\nu,\nu') \in \mathcal{E}} \frac{1}{Z} \phi(\nu, \nu')$, where Z is called a partition function, and ϕ are so-called compatibility functions. This suggests that an algorithm that solves local subproblems could be used. One possible choice is the variable elimination algorithm, which is an exact standard algorithm suitable for undirected graphs of moderate size. To make the computation faster, satGP currently modifies it by computing each diagonal in the graph, shown in Fig. 2, in parallel from $\nu^{0,0}$ to $\nu^{n_{\text{lat}}, n_{\text{lon}}}$ and then back from $\nu^{n_{\text{lat}}, n_{\text{lon}}}$ to $\nu^{0,0}$. Each ν_{ij} is conditioned on the previously evaluated vertices in $\partial\nu^{ij}$, but the diagonal edges of the so-called reconstituted graph are not introduced, as would normally be done. When starting again from the bottom-right corner after computing diagonals numbered $1 \ldots N$, the $(N+1)$th diagonal is not conditioned on previously computed nodes. Once the diagonals n_{lon} and $N + n_{\text{lon}} - 2$ that "sandwich" the node ν from both upper-left and lower-right sides have been computed, the posterior distribution of $\boldsymbol{\beta}_\nu$ – and any other vertex on the $(N + n_{\text{lon}} - 1)$th diagonal – can be calculated.

The modification of the algorithm loses the ability of the upper-right and lower-left corners to communicate effectively, but since most remote sensing data sets contain at least some observations for some time period for most nodes, the far-away information does not affect results in many practical scenarios. Techniques such as generalized belief propagation (Wainwright and Jordan, 2008) could be used to obtain a better fit to the data, in case a need emerges to improve the spatial fitting of the mean function coefficients.

The results should not change due to changes in the user-chosen grid resolution, and for this reason satGP inversely weights the edges exponentially according to the distances between the (geographical) coordinates corresponding to the connected nodes. This rate of exponential decay is user configurable by the `dscale` parameter; see Appendix A.

2.3.2 Computing the individual posterior marginals $p(\boldsymbol{\beta}_\nu | \boldsymbol{\psi}^{\text{obs}}, \boldsymbol{\theta})$

Assume that for the vertex ν in Fig. 2 the neighbors marked ∂_ν have been computed. Computing the marginal distribution of $\boldsymbol{\beta}$ and an estimate of $\boldsymbol{\delta}$ at ν, referred to below as $\boldsymbol{\beta}_\nu$ and $\boldsymbol{\delta}_\nu$, is carried out in several steps. These steps take place inside solving the spatial problem described above as follows: the steps listed below are computed for each vertex, corresponding to a spatial location. The computation uses information from previously computed points as prior information.

In the particular form of the mean function m used for OCO-2 data in Eq. (11), the phase-shift parameter $\boldsymbol{\delta}$ cannot be estimated with regression the way $\boldsymbol{\beta}$ is found in Eqs. (9) and (10). For this reason, the nonlinear space-dependent $\boldsymbol{\delta}$ parameters are found with an optimization algorithm from the NLopt package (Johnson, 2014), by default the BFGS (Broyden–Fletcher–Goldfarb–Shanno) algorithm, before finding $\widehat{\boldsymbol{\beta}}$ with Eq. (6). After obtaining $\widehat{\boldsymbol{\beta}}$ the $\boldsymbol{\delta}$ parameters are re-optimized given the $\widehat{\boldsymbol{\beta}}$. The full calibration process for a single graph node ν proceeds in the following manner:

1. Select n_ν observations $\boldsymbol{\psi}_\nu^{\text{obs}}$ of the observable that are close in terms of the spatial components of the covariance. Specifically, when evaluating whether to select an observation $\boldsymbol{\psi}_i^{\text{obs}}$ for carrying out computations at test input $\boldsymbol{x}^*$ corresponding to some vertex ν, we set the time component of $\mathbf{x}_i^{\text{obs}}$ to that of the test input, $x_i^{\text{obs t}} \leftarrow x^{*\text{t}}$, making the temporal part of the covariance function irrelevant in this selection process. Observation selection is described in detail in Sect. 2.5.

2. Find a best-guess $\boldsymbol{\delta}_\nu$ (and $\boldsymbol{\beta}_\nu$, which is not used) by running the BFGS optimization algorithm (Nocedal, 1980) to find an approximate maximum a posteriori estimate by computing

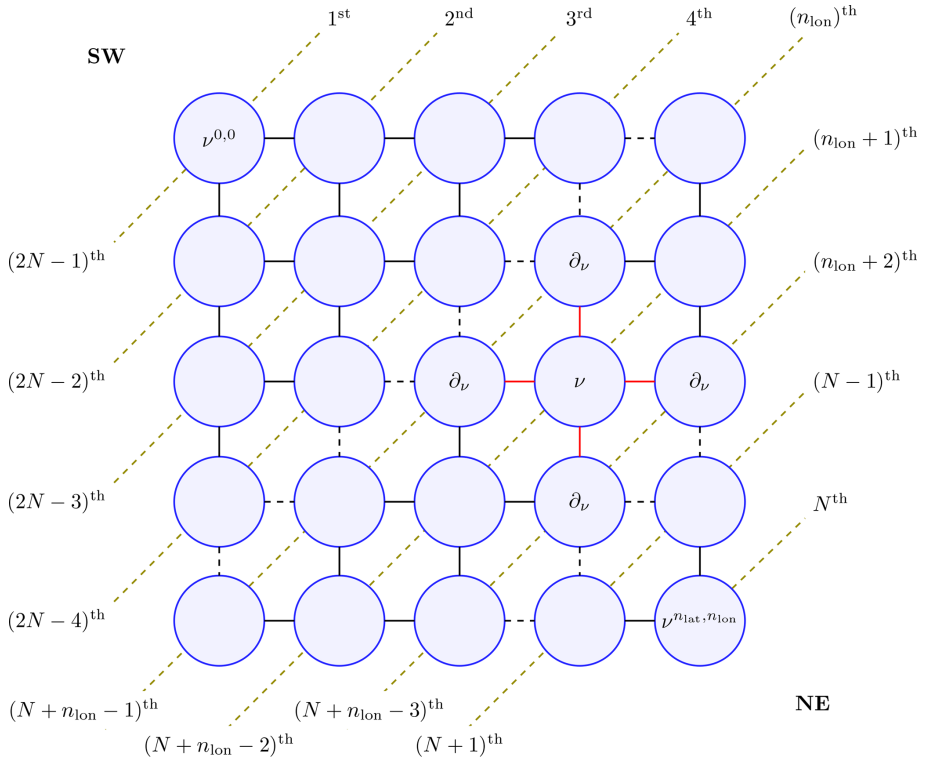

Figure 2. The marginal distribution $p(\nu)$ of vertex ν is conditional only on the neighbors in ∂_ν (connected to ν with red edges) due to the Markov structure in the pictured lattice graph. For effective solving, the vertices on the diagonal dashed lines are computed simultaneously making the algorithm non-exact. The order numbers labeling the diagonal lines represent an ordering in which the diagonals can be computed in parallel to get all the marginals in $\mathcal{O}(N)$ wall time, where $N \triangleq n_{\text{lat}} + n_{\text{lon}} - 1$. Southwest and northeast corners of the domain are labeled SW and NE in the graph. The final values of the parameters are obtained when diagonals from N to $2N-1$ are computed.

$$\widetilde{\boldsymbol{\beta}}_\nu, \widetilde{\boldsymbol{\delta}}_\nu = \arg\min_{\boldsymbol{\beta},\boldsymbol{\delta}} \Bigg\{ \sum_{j=1}^{n_\nu} (m(\boldsymbol{x}_\nu;\boldsymbol{\beta}_\nu,\boldsymbol{\delta}_\nu) - \boldsymbol{\psi}_\nu^{\text{obs}}{}_j)^2 + \sum_{\nu' \in \partial\nu} \Big((\boldsymbol{\delta}_\nu - \boldsymbol{\delta}_{\nu'})^T (\boldsymbol{\delta}_\nu - \boldsymbol{\delta}_{\nu'}) + (\boldsymbol{\beta}_\nu - \boldsymbol{\beta}_{\nu'})^T (\boldsymbol{\beta}_\nu - \boldsymbol{\beta}_{\nu'}) \Big) \Bigg\}. \tag{12}$$

The first sum runs over the training data selected by the observation selection method described in Sect. 2.5, and the second sum constrains the parameter values close to those in ∂_ν. This optimization problem is very simple since there are few $\boldsymbol{\beta}$ and/or $\boldsymbol{\delta}$ parameters for the individual vertices.

3. Given $\widetilde{\boldsymbol{\delta}}_\nu$, an estimate of the GP covariance parameters $\widetilde{\boldsymbol{\theta}}$ – e.g., from a previous simulation or a best guess – and the observations $\boldsymbol{\psi}_\nu^{\text{obs}}$, compute $\mathbb{E}[\boldsymbol{\beta}_\nu | \boldsymbol{\psi}_\nu^{\text{obs}}, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu]$ and $\text{Cov}[\boldsymbol{\beta}_\nu | \boldsymbol{\psi}_\nu^{\text{obs}}, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu]$ via Eqs. (6) and (7). Together these give $p(\boldsymbol{\beta}_\nu | \boldsymbol{\psi}_\nu^{\text{obs}}, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu)$. If this computation uses a flat prior, as we do in this work, this is by Bayes' theorem proportional to the likelihood $p(\boldsymbol{\psi}_\nu^{\text{obs}} | \boldsymbol{\beta}_\nu, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu)$.

4. Find the posterior marginal distribution of $\boldsymbol{\beta}_\nu$ by applying Bayes' theorem and using the computed distributions at the neighboring nodes as prior information. Due to the Markov structure this becomes $p(\boldsymbol{\beta}_\nu | \boldsymbol{\psi}_\nu^{\text{obs}}, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu) \propto p(\boldsymbol{\psi}_\nu^{\text{obs}} | \boldsymbol{\beta}_\nu, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_\nu) \prod_{\nu^{ij} \in \partial_\nu} p(\boldsymbol{\beta}^{ij} | \boldsymbol{\psi}_{\nu^{ij}}^{\text{obs}}, \widetilde{\boldsymbol{\theta}}, \widetilde{\boldsymbol{\delta}}_{\nu^{ij}})$. If the spatial location corresponding to ν does not have any data to inform the fit (if $\boldsymbol{\psi}_\nu^{\text{obs}}$ is a zero-length vector), then parameter values from ∂_ν will determine the fit.

5. Using the $\boldsymbol{\beta}_\nu$ obtained at the previous step, re-optimize *only* the $\boldsymbol{\delta}_\nu$ parameters as above in step number 2. Since $\boldsymbol{\beta}_\nu$ is not varied, the term $(\boldsymbol{\beta}_\nu - \boldsymbol{\beta}_{\nu'})^T (\boldsymbol{\beta}_\nu - \boldsymbol{\beta}_{\nu'})$ in Eq. (12) plays no role here.

The mean value of the distribution of $\boldsymbol{\beta}_\nu$ coming out from step 4 corresponds to the $\widehat{\boldsymbol{\beta}}$ in, e.g., Eq. (9), where $\boldsymbol{x}^*$ would now refer to the spatial location of vertex ν. Similarly, in case $\boldsymbol{\delta}$-type coefficients are used, the functions f_i will depend on the final $\boldsymbol{\delta}_\nu$ values computed in step 5. The full sets of $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ coefficients for all the vertices in the graph are denoted by $\boldsymbol{\beta}_\mathcal{V}$ and $\boldsymbol{\delta}_\mathcal{V}$, and the sets of calibrated values are written as $\widehat{\boldsymbol{\beta}}_\mathcal{V}$ and $\widehat{\boldsymbol{\delta}}_\mathcal{V}$.

2.4 Covariance functions in satGP

The smoothness, amplitude, and length scales of the Gaussian process realizations are determined by the covariance kernel used. The satGP program supports several different types of covariance function components for forming the full covariance function k in Eq. (1). The options available reflect the properties that can be expected in remote sensing data – varying smoothness and meridional and zonal length scales, potential periodicity, and changing the orientation of the data-informed and uninformed axes according to wind speed and direction. This section lists the available covariance function formulations, and other forms may be easily added in the code.

For convenience, let

$$\xi_I^\gamma(\boldsymbol{x},\boldsymbol{x}') \triangleq \sum_{c\in I}\left|\frac{x^c - x'^c}{\ell_c}\right|^\gamma = \|\mathbf{P}^I(\boldsymbol{x}) - \mathbf{P}^I(\boldsymbol{x}')\|_{\boldsymbol{\Gamma}}^\gamma, \tag{13}$$

where $\gamma > 0$ is the exponent, I is a (sub)set of the dimensions of the inputs $\boldsymbol{x}$ and $\boldsymbol{x}'$, ℓ_c values are length-scale parameters corresponding to the different dimensions in I, e.g., temporal (ℓ_t), zonal (ℓ_lon), or meridional (ℓ_lat) directions, and x^c values are different components of the inputs, e.g., x^t, x^lat, and x^lon. The spatial length-scale parameters ℓ_lat and ℓ_lon are in units of distance on the surface of the unit sphere, corresponding to radians at the Equator. The $\mathbf{P}^I$ matrix projects $\boldsymbol{x}$ onto indices/dimensions in I; $\boldsymbol{\Gamma}$ is a diagonal covariance matrix with diagonal elements ℓ_c^2, and the notation $\|\boldsymbol{r}\|_{\boldsymbol{\Gamma}}$ stands for $\sqrt{\boldsymbol{r}^T\boldsymbol{\Gamma}^{-1}\boldsymbol{r}}$, where $\boldsymbol{r}$ is an arbitrary vector of the appropriate size. For remote sensing data used in this work, space-only variables form the set $I_\mathrm{S} \triangleq \{\mathrm{lat}, \mathrm{lon}\}$, and for spatial and temporal variables together the notation $I_\mathrm{ST} \triangleq \{\mathrm{lat}, \mathrm{lon}, \mathrm{t}\}$ is used. Notation lat and lon refer to the spatial components of $\boldsymbol{x}$, collectively earlier referred to as $\boldsymbol{x}^\mathrm{s}$, and t refers to the temporal component. The form of ξ in Eq. (13) implies that the different dimensions have separate length-scale parameters ℓ. The exponent γ in ξ is, however, shared between the dimensions. For the set of all ℓ parameters over a set I' of dimensions we write $\boldsymbol{\ell}_{I'}$. All the covariance functions below depend on a parameter τ, square of which determines the maximum covariance that is attained when $\boldsymbol{x} = \boldsymbol{x}'$.

The exponential family of covariance functions with parameters $\boldsymbol{\theta}_\mathrm{exp} \triangleq [\tau, \boldsymbol{\ell}_{I_\mathrm{ST}}, \gamma]^T$ is defined by the covariance function

$$k_\mathrm{exp}(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}_\mathrm{exp}) \triangleq \tau^2 \exp\left(-\xi_{I_\mathrm{ST}}^\gamma(\boldsymbol{x},\boldsymbol{x}')\right). \tag{14}$$

The exponent γ controls the smoothness of the samples from the Gaussian process, with $\gamma = 2$ yielding infinitely differentiable realizations.

The Matérn family of covariance functions, with $\boldsymbol{\theta}_\mathrm{M} \triangleq [\tau, \boldsymbol{\ell}_{I_\mathrm{ST}}, \nu]^T$ is given by the covariance

$$k_\mathrm{M}(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}_\mathrm{M}) \triangleq \frac{\tau^2 s^\nu}{\boldsymbol{\Gamma}(\nu)2^{\nu-1}} K_\nu(s), \tag{15}$$

where $s = 2\sqrt{\nu}\xi_{I_\mathrm{ST}}^1(\boldsymbol{x},\boldsymbol{x}')$ and ν controls the smoothness parameter usually denoted by α via $\alpha = \nu + \frac{q}{2}$. The function K_ν is the modified Bessel function of the second kind of order ν. With $q = 1$, the value $\nu = \infty$ corresponds to the squared exponential kernel and $\nu = 0.5$ to the exponential kernel with $\gamma = 1$. Despite this similarity between the Matérn and exponential kernels, the realizations of the random function from the processes with values $\frac{1}{2} < \nu < \infty$ do not correspond to those with the kernel k_exp with any value of γ.

A periodic kernel with $\boldsymbol{\theta}_\mathrm{per} \triangleq [\tau, \boldsymbol{\ell}_{I_\mathrm{S}}, \ell_\mathrm{per}]^T$ is defined in satGP by

$$k_\mathrm{per}(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}_\mathrm{per}) \triangleq \tau^2 \exp\left(-\frac{2}{\ell_\mathrm{per}^2}\sin^2\left(\pi\left[\frac{x^\mathrm{t} - x'^\mathrm{t}}{\Delta_\mathrm{period}}\right]\right) - \xi_{I_\mathrm{S}}^2(\boldsymbol{x},\boldsymbol{x}')\right). \tag{16}$$

The parameter Δ_period is the period length, which is assumed to be well known a priori and therefore is not among the parameters that are calibrated. The second term in the exponent controls the spatial dependence via length-scale parameters in $\boldsymbol{\ell}_{I_\mathrm{S}}$, and ℓ_per determines how far the temporal covariance extends, modulo Δ_period.

satGP contains an additional covariance function that utilizes local wind information when computing the covariances. The underlying rationale is that winds affect how quantities of interest such as gases in the atmosphere or algae blooms in surface water spread. For this reason, if wind data are available, it is natural to try to use them for inference with the Gaussian process. We define the wind-informed covariance kernel with parameters $\boldsymbol{\theta}_\mathrm{w} \triangleq [\tau, \ell, \ell_\mathrm{t}, \rho, \boldsymbol{w}^*]^T$ by

$$k_\mathrm{w}(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}_\mathrm{w}) \triangleq k_\mathrm{exp}(\boldsymbol{x}_\mathrm{w},\boldsymbol{x}'_\mathrm{w};\tau,\{\ell_\parallel,\ell_\perp,\ell_\mathrm{t}\},2). \tag{17}$$

The parameter ρ in $\boldsymbol{\theta}_\mathrm{w}$ defines how strongly the magnitude of the wind vector at the test input, $\boldsymbol{w}^* \triangleq [w^*_\mathrm{lat}, w^*_\mathrm{lon}]^T$ (the last parameter in $\boldsymbol{\theta}_\mathrm{w}$), affects the shape of the covariance. The kernel itself is an exponential kernel, where the spatial components of the vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ are transformed by wind data, and the covariance lengths are transformed by wind speed. A spatiotemporal vector $\boldsymbol{x} = [x^\mathrm{lat}, x^\mathrm{lon}, x^\mathrm{t}]$ is transformed by wind to the vector $\boldsymbol{x}_\mathrm{w}$ in a new coordinate system according to

$$\boldsymbol{x}_\mathrm{w} \triangleq \begin{pmatrix} (\boldsymbol{x}^\mathrm{s} - \boldsymbol{x}^{*\mathrm{s}})^T\boldsymbol{w}^\parallel \\ (\boldsymbol{x}^\mathrm{s} - \boldsymbol{x}^{*\mathrm{s}})^T\boldsymbol{w}^\perp \\ x^\mathrm{t} \end{pmatrix}, \tag{18}$$

where $\boldsymbol{x}^\mathrm{s}$ and $\boldsymbol{x}^{*\mathrm{s}}$ are the spatial components of vectors $\boldsymbol{x}$ and $\boldsymbol{x}^*$, and $\boldsymbol{w}^\parallel$ and $\boldsymbol{w}^\perp$ are the unit vectors in the lat–long coordinates along and perpendicular to wind direction at the test input $\boldsymbol{x}^*$.

The spatial scaling (ℓ) parameters for k_w, corresponding now to the covariance scales parallel and perpendicular to the wind direction, are given by

$$\ell_\parallel \triangleq \ell\sqrt[4]{1 + |\boldsymbol{w}^*|\rho}, \qquad \ell_\perp \triangleq \ell. \tag{19}$$

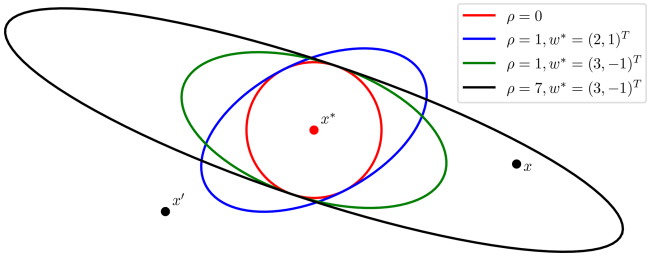

Figure 3. Equicovariance ellipses from the wind-informed kernel with various wind vectors $\boldsymbol{w}^*$ and values of ρ. The wind values are taken at the test input $\boldsymbol{x}^*$, but the covariance function k is evaluated also for each pair of observations $\boldsymbol{x}$ and $\boldsymbol{x}'$.

The parameter vector for the exponential kernel then becomes $\boldsymbol{\theta}_{\text{exp}} \leftarrow [\tau, \ell_{\parallel}, \ell_{\perp}, \ell_t, 2]^T$, where the last element denotes the exponent γ used by the exponential kernel. A number of possible covariance ellipses resulting from the transformation procedure are shown in Fig. 3. Some data sets, like OCO-2, incorporate wind information, and satGP does have the capability of gridding that data using another Gaussian process. Reading in gridded wind data from other sources is also a possibility. Using k_{w} requires that wind data are available at each $\boldsymbol{x}^*$.

The covariance functions used in this work to model Ψ are sums of several kernels – sums of valid Gaussian process kernels remain valid kernels. The general form of the *multi-scale kernel* used in satGP is given by

$$k\left(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\right) = \quad \delta_{x,x'}\sigma_x^2 + \sum_{i=1}^{n_{\text{ker}}} k_{\text{ker}_i}\left(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}_{\text{ker}_i}\right), \tag{20}$$

where the first term, which in kriging is called the nugget, contains the observation error variances, and each $\text{ker}_{\text{i}} \in \{\text{exp}, M, \text{per}, \text{w}\}$.

The kernel components of a multi-scale kernel are in this work called *subkernels*. The combined set of parameters is denoted by $\boldsymbol{\theta} = [\boldsymbol{\theta}_{\text{ker}_1}^T, \ldots, \boldsymbol{\theta}_{\text{ker}_{n_{\text{ker}}}}^T]^T$. Not all subkernel types are included in all experiments – rather, the simulations in Sect. 4 utilize kernels with one to three components. What those components should be depends on what fields are being modeled and what kinds of correlation structures the user expects to find in the data. Section 4.1 discusses identifiability of the different subkernel parameters of the multi-scale kernel.

Instead of calling $k(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta})$ in Eq. (20) a multi-scale kernel, the term multi-component kernel could also be used to describe the form. The term "multi-scale" underlines that the purpose of the combined kernel is to model data well, which contains several natural length scales, as remote sensing products often do. Furthermore, we believe that combining several kernels with identical length-scale parameters does not represent a common use case.

2.5 Covariance localization and observation selection for the multi-scale kernel

Using a large number of observations makes solving the Gaussian process Eqs. (9) and (10) intractable as the cost of inverting the covariance matrix scales as $\mathcal{O}(n_{\text{obs}}^3)$. This creates a need for finding approximate solutions while introducing as little error as possible. In satGP, covariance localization is used to utilize only a subset of observations for computing Eqs. (9) and (10). To control the localization behavior the user needs to set two parameters: the maximum subkernel covariance matrix size κ and the minimum covariance parameter σ_{min}^2.

Assume that the multi-scale kernel defined by the user contains n_{ker} subkernels. For each test input $\boldsymbol{x}^*$ and for each subkernel k_l, the set of observations feasible for inclusion in $\mathbf{K}$ in Eqs. (6) and (7) is

$$A_{*,l}^{\text{obs}} \triangleq \Big\{ \psi_i \in \boldsymbol{\psi}^{\text{obs}} | k_l(\boldsymbol{x}_i^{\text{obs}}, \boldsymbol{x}^*) > \sigma_{\text{min}}^2, \\ \psi_i \notin A_{*,j}^{\text{obs}} \ \forall j < l \Big\}, \tag{21}$$

where the last condition prevents observations from being added by several subkernels. In the end we select a single set of observations A_*^{obs} for each test input by combining some or all of the observations included in each $A_{*,l}^{\text{obs}}$. The observation selection proceeds sequentially through the list of subkernels according to the procedure presented in Fig. 4. Recomputing the κ' for each subkernel on line 3 of the algorithm allows the selection of more than κ observations by subkernels if the previous subkernels did not have κ feasible observations available. This is done to allow the full kernel size to grow to $n_{\text{ker}}\kappa$ when possible. On line 4, the observation selection operator $\mathcal{S}(A_{*,l}^{\text{obs}}, \kappa')$ chooses κ' observations from each $A_{*,l}^{\text{obs}}$ either greedily by picking the observations with highest covariance with $\boldsymbol{x}^*$ or randomly by sampling uniformly without replacement from $A_{*,l}^{\text{obs}}$. Out of these two methods random selection avoids observation sorting and is therefore faster, especially if a huge number of data are near the test input $\boldsymbol{x}^*$. This comes at the cost of producing noisier fields of marginal posterior means. For covariance parameter estimation random selection works well. See Appendix A for additional details.

Since the subkernels are handled sequentially, their order may affect which observations are selected due to the exclusion in Eq. (21), and to grow the full kernel to size $n_{\text{ker}}\kappa$ as often as possible, it is recommended to specify the subkernel with the largest ℓ parameters as the last one. After constructing A_*^{obs}, the covariance matrix $\mathbf{K}$ is constructed by evaluating the full covariance function k according to Eq. (20) for all pairs of selected observations.

For learning the spatially varying $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ parameters for grid index (i, j) in the mean function with the methods in Sect. 2.3.2, the observation selection is performed by disregarding the time component on the inputs, i.e., by setting

Data: Set of feasible observations $A^{\text{obs}}_{*,l}$ for each subkernel, maximum subkernel size κ, observation selection operator $\mathcal{S}$.
Result: Set A^{obs}_{*} of observations that are informative for test input x^*

$A^{\text{obs}}_{*} \leftarrow \emptyset$;
for $i \leftarrow 1$ **to** n_{ker} **do**
  $\kappa' \leftarrow i\kappa - |A^{\text{obs}}_{*}|$;
  $A^{\text{obs}}_{*} \leftarrow A^{\text{obs}}_{*} \cup \mathcal{S}(A^{\text{obs}}_{*,i}, \kappa')$;
end

Figure 4. Algorithm for selecting observations for carrying out predictions at test input $\boldsymbol{x}^*$. The sets A^{obs}_{*} are defined by Eq. (21), and the variable κ is the maximum subkernel size, also listed in Table 2 and discussed in Sect. 3. The selection operator $\mathcal{S}(A^{\text{obs}}_{*,l}, \kappa')$ chooses κ' observations from each $A^{\text{obs}}_{*,l}$ either greedily or randomly.

$\mathbf{x}^{\text{obs}\,\text{t}}_i \leftarrow \boldsymbol{x}^{ij\,\text{t}}$ for all $\boldsymbol{x}^{\text{obs}}_i$ in the training data. The reason for this is that, since learning the mean function amounts to fitting spatially varying parameter vectors $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$, the data to perform the fit should not be selected based on covariance in the time direction, as the mean function should be equally valid at all times.

Selecting the observations could also be done based on values of k instead of each k_l individually or by other approaches, such as the one presented by Schäfer et al. (2017). However, even though the method of observation selection does have an effect on the inferred posterior marginals, the screening property of Gaussian processes ensures that this effect is not major as long as observational noise is small and the nearest observations are included in all directions. The parameter identifiability results in Sect. 4.1 and the WACCM4 results in Sect. 4.2 verify that the current nearest-neighbor-in-covariance approach works as intended.

2.6 Learning the covariance parameters $\boldsymbol{\theta}$

From Sect. 2.1 the log marginal likelihood of observations $\boldsymbol{\psi}^{\text{obs}}$ given a set of parameters $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and $\boldsymbol{\delta}$ is given by

$$2\log p\left(\boldsymbol{\psi}^{\text{obs}}|\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\theta}\right) = \\ -\|\left(\boldsymbol{\psi}^{\text{obs}} - \mathbf{F}\boldsymbol{\beta}\right)\|^2_{\mathbf{K}} - \log|\mathbf{K}| - n_{\text{obs}}\log(2\pi), \quad (22)$$

where the covariance function parameters $\boldsymbol{\theta}$ implicitly determine $\mathbf{K}$, and the nonlinear space-dependent mean function parameters $\boldsymbol{\delta}$ affect the values in $\mathbf{F}$. The maximum (marginal) likelihood estimate (MLE) $\widehat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$ can be found via minimizing

$$\mathcal{L}(\boldsymbol{\theta}) = \|\left(\boldsymbol{\psi}^{\text{obs}} - \mathbf{F}\widehat{\boldsymbol{\beta}}\right)\|^2_{\mathbf{K}} + \log|\mathbf{K}| + n_{\text{obs}}\log(2\pi) \quad (23)$$

as stated in context of Eq. (5).

In the presence of a huge number of observations, calculating the determinant of the full covariance $|\mathbf{K}|$ is not feasible, and maximizing the log-likelihood is approximated by

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{\boldsymbol{x}^*_i \in E_{\text{ref}}} \left\{\|\left(\boldsymbol{\psi}^{\text{obs}}_{\text{local}_i} - \mathbf{F}_i\widehat{\boldsymbol{\beta}}\right)\|^2_{\widetilde{\mathbf{K}}_i} + \log|\widetilde{\mathbf{K}}_i|\right\}, \quad (24)$$

where E_{ref} is a set of randomly sampled points from the spatiotemporal domain specified for the experiment, determined by the parameters area and n_{days} in Table 2. The $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\delta}}$ parameters, the latter of which is embedded in $\mathbf{F}$, are the point estimates corresponding to each $\boldsymbol{x}^*_i$, interpolated from the values obtained for the full grid. The optimization in Eq. (24) is carried out over all subkernel parameters with some caveats: currently the smoothness-related parameter ν for the Matérn kernel and the exponent γ for the exponential kernel are not calibrated, and naturally neither are the wind data $\boldsymbol{w}^*$ listed as a parameter for the wind-informed covariance – however, the parameter ρ affecting that kernel can be learned.

While the selection of inputs included in E_{ref} has an effect on the obtained parameter estimate, that effect has proven in simulations to be small. The vectors $\boldsymbol{\psi}^{\text{obs}}_{\text{local}_i} \in \mathbb{R}^{d_i}$, where d_i is the number of observations chosen by the observation selection method of Sect. 2.5 for test input $\boldsymbol{x}^*_i$, contain observations closest in covariance to $\boldsymbol{x}^*_i$, each of which is a reference point included in E_{ref}. The matrices $\mathbf{F}_i$ are the corresponding $\mathbf{F}$-matrices, as described in Sect. 2.1. The last term in Eq. (23) is dropped, since while varying $\boldsymbol{\theta}$ in Eq. (24) changes d_i, the number of total observations in the problem should fundamentally stay the same.

The maximum likelihood estimate approximation in Eq. (24) contains a sum over blocks of observations, which can together be thought of as a block-diagonal approximation of the full dense covariance for all observations in all $\boldsymbol{\psi}^{\text{obs}}_{\text{local}_i}$. The blocks in this approximation are the dense covariance matrices $\widetilde{\mathbf{K}_i}$, and in contrast to a full dense $\mathbf{K}$, in this approximation the cross-covariances between observations in $\boldsymbol{\psi}^{\text{obs}}_{\text{local}_i}$ and $\boldsymbol{\psi}^{\text{obs}}_{\text{local}_j}$, $i \neq j$, are set to 0. This is done even if the randomly selected corresponding inputs $\boldsymbol{x}^*_i$ and $\boldsymbol{x}^*_j$ are close to each other. Due to the $\mathcal{O}(n^3)$ cost of inverting the covariance matrix, which is needed for finding the maximum likelihood estimate, using the block approximation provides a critical efficiency improvement without which learning the covariance function parameters would not be feasible.

While this method is suitable for finding point estimates for the parameters $\boldsymbol{\theta}$, the computed approximated log-likelihood has an unknown scaling factor resulting in an unknown multiplicative factor for the variance term in the exponent of the Gaussian distribution, and hence information about the true size of the posterior distribution of the covariance parameters $p(\boldsymbol{\theta}|\boldsymbol{\psi}^{\text{obs}}, \widehat{\boldsymbol{\beta}}_{\mathcal{V}}, \widehat{\boldsymbol{\delta}}_{\mathcal{V}})$ is lost.

By default the scaled posterior $p(\boldsymbol{\theta}|\boldsymbol{\psi}^{\text{obs}}, \hat{\boldsymbol{\beta}}_{\mathcal{V}}, \hat{\boldsymbol{\delta}}_{\mathcal{V}})$ is explored by using the adaptive Metropolis (AM) Markov chain Monte Carlo (MCMC) algorithm (Haario et al., 2001), an im-

Table 2. Most important satGP control variables and high-level C structs: first section contains parameters for program logic, second for domain specification, third for covariance and mean function definition, and last for observation handling. This list is by no means exhaustive – the configuration file contains lots of variables that can control the program. Some additional tweaking is possible by changing hard-coded values directly in the source code, such as those listed in Appendix A.

Variable	Type	Low	High	Notes
learn_k	`int`	0	2	(0) Do not train $\boldsymbol{\theta}$; (1) generate observations and learn $\boldsymbol{\theta}$; (2) learn $\boldsymbol{\theta}$ from non-synthetic data.
learn_m	`int`	0	1	(0) Do not train local $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$; (1) find local $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ as in Sect. 2.3.
sampling	`int`	0	2	(0) Skip sampling; (1) calculate GP marginals at each grid point; (2) sample from GP.
area	`char*`	–	–	Area definition setting longitude and latitude minimum and maximum values
n_{days}	`int`	1	∞	Number of days to be simulated
ω	`float`	> 0	180	1 d grid resolution in degrees – small values degrade esp. posterior sampling performance.
n_{ker}	`int`	1	10	Number of subkernels k_l in k
cfc	`struct*`	–	–	Recursive struct pointer defining $k_1 \ldots k_{n_{\text{ker}}}$ and corresponding $\boldsymbol{\theta}$; see Sect. 2.4.
mf	`struct*`	–	–	Struct pointer for defining type of $m(\cdot,\cdot)$ and associated (initial) $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$; see Sect. 2.2.
ζ_{train}	`float`	0	∞	Fraction of observations that are randomly included in $\boldsymbol{\psi}^{\text{obs}}$ when learning $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, and $\boldsymbol{\delta}$
ζ_{sample}	`float`	0	∞	Fraction of observations that are randomly included in $\boldsymbol{\psi}^{\text{obs}}$ when sampling$\neq 0$
σ^2_{min}	`float`	0	∞	Discard observation at $\boldsymbol{x}_i^{\text{obs}}$ for $\boldsymbol{x}^*$ if $k(\boldsymbol{x}_i^{\text{obs}}, \boldsymbol{x}^*) < \sigma^2_{\text{min}}$; see Sect. 2.5.
n_{ref}	`int`	0	∞	Number of reference points in E_{ref} in Eq. (24) for training $\boldsymbol{\theta}$
$n_{\text{synthetic}}$	`int`	0	∞	Number of random locations where synthetic data are generated for training $\boldsymbol{\theta}$
$\sigma^2_{\text{synthetic}}$	`float`	0	∞	Variance in Gaussian noise added to synthetic observations
κ	`int`	1	∞	Maximum subkernel size; values $\kappa > n_{\text{ker}}^{-1}1000$ will be slow due to $\mathcal{O}(\kappa^3)$ scaling.

plementation of which is included in the satGP source code. MCMC methods (Gamerman, 1997) are used to draw samples from probability distributions when direct sampling is not possible, but the likelihood function can still be evaluated. The samples are drawn by generating a Markov chain of parameter values, which is an autocorrelated sample from the posterior. The AM algorithm is an adaptive method that is efficient for many real-world sampling situations. The observation selection procedure in Sect. 2.5 introduces discontinuities to the posterior distribution due to selected observations changing when the covariance function parameters are modified. Computing $\hat{\boldsymbol{\theta}} \leftarrow \mathbb{E}[\boldsymbol{\theta}|\boldsymbol{\psi}^{\text{obs}}, \hat{\boldsymbol{\beta}}_{\mathcal{V}}, \hat{\boldsymbol{\delta}}_{\mathcal{V}}]$ with MCMC – i.e., using the posterior mean of a Monte Carlo sample – usually works around this noisiness in the likelihood. On the downside, MCMC is computationally much more demanding than finding the maximum a posteriori estimate with optimization, since MCMC may require computing up to millions of likelihood evaluations. In the satGP context using MCMC is feasible since the forward model simply amounts to sampling from a multivariate normal distribution, which is very fast. Furthermore, the parameter dimension is moderate, even with multiple subkernels, limiting the need to generate extremely long chains. The current version of satGP uses a flat prior distribution for the covariance parameters, with hard limits on the parameter ranges.

The software also includes a capability to learn the covariance parameters using optimization algorithms such as COBYLA or SBPLEX available in NLopt. These methods are much faster than MCMC but have the tendency of getting stuck in local minima, limiting their usefulness.

3 Overview of Computation

The satGP code is written in C, with visualization scripts written in Python and parallelization implemented with OpenMP directives. The program reads data from netCDF and text files and the configuration from a C header file. For linear algebra satGP uses the C interfaces of LAPACK and BLAS and LAPACKE and CBLAS, and optimization tasks are carried out with the NLopt library. The computations are performed in single precision in order to save memory resources with the largest data sets and also to improve performance.

The most important configuration variables are listed in Table 2. The user needs to define whether parameters are learned or prescribed and whether marginals or samples from the GP are to be computed. The mean function and the covariance kernel are defined by initializing corresponding structs with parameters and their limits if calibration is to be performed. For computing GP marginals or drawing samples from the random process, the geographic and temporal extents need to be specified, and the mean function and the covariance kernel used must be given.

Several parameters can be tweaked to improve computational efficiency, including all of those in the second and last sections of Table 2. The first main bottleneck for computing a marginal at $\boldsymbol{x}^*$ is sorting the observations for selecting the

most informative ones to be used in the covariance matrices; see Sect. 2.5. This requires roughly $\mathcal{O}(r_l \log r_l + \kappa \log \kappa)$ operations for each subkernel, where r_l is the number of grid locations $\boldsymbol{x}^{ij}$ in the spatiotemporal grid such that for the lth subkernel, $k_l(\boldsymbol{x}^{ij}, \boldsymbol{x}^*) > \sigma^2_{\min}$. For subkernels with $\gamma = 2$, $r_l \propto \prod_{i=1}^{q} \ell_i^l$, with ℓ_i^l denoting the length-scale parameters over all the dimensions of the inputs $\boldsymbol{x}$. In other words, r_l is proportional the size of the hypersphere inside which observations are considered for each $\boldsymbol{x}^*$. The second bottleneck is calculating the Cholesky decompositions of the covariance matrices $\mathbf{K}$ with cost $\mathcal{O}((n_{\ker}\kappa)^3)$. The cost of calculating the means and variances for the GP in a grid for a set of n_{times} points on the time axis is therefore given by

$$\text{cost} = \mathcal{O}\left(\frac{A n_{\text{times}}}{\omega^2}\left[(n_{\ker}\kappa)^3 + \sum_{l=1}^{n_{\ker}} (r_l \log r_l + \kappa \log \kappa)\right]\right), \tag{25}$$

where A is the grid area in degrees squared, and ω is the grid resolution. When the random observation selection method mentioned in Sect. 2.5 is used, the $r_l \log r$ in Eq. (25) becomes just r_l.

The execution of the program is presented in Fig. 5. The function `AddToState()` reads observations (asynchronously) into a state object that tracks the proximity of each observation to each grid point. Only a part of the observations is added, controlled on line 6 by the parameter η^i_{train}, which corresponds to the inclusion probability of each observation. This probability depends on ζ_{train} in Table 2 via

$$\eta^i_{\text{train}} \triangleq \frac{d\left(\mathbf{x}_i^{\text{obs}}, \mathbf{x}_{i_{\text{prev}}}^{\text{obs}}\right)}{\omega \zeta_{\text{train}}} \wedge 1, \tag{26}$$

where $d(\mathbf{x}_i^{\text{obs}}, \mathbf{x}_{i_{\text{prev}}}^{\text{obs}})$ is the Euclidean distance of the point at $\mathbf{x}_i^{\text{obs}}$ that is being proposed for addition to the previous added point at $\mathbf{x}^{\text{obs}}_{\text{prev}_i}$, and $\wedge$ is the standard notation for minimum. Hence with $\zeta = 0$ all observations are added.

For computing the marginals, the spatial domain can be decomposed with `Decompose()`, line 23, into several spatial subdomains (sd) so that arbitrary-size grids can be computed. This makes solving large problems with limited amount of memory possible, but it only works with sampling$= 2$. This option is in practice rarely needed, and it was not needed for the simulations in Sect. 4. The state object is emptied by `ReInitializeState()`, which also potentially sets new subdomain extents. Function `SampleFromPrior()` actually performs the computations on ll. 30–37 but with the inputs $\boldsymbol{x}^*$ in a random pattern instead of in a grid as is the case in ll. 27–38.

The `AddSubdomainData()` method on l. 29 adds data as on ll. 3–9 but only to the current subdomain. After that, the `SelectObservations()` method (l. 31) carries out selecting the best observations as described in Sect. 2.5.

Data: filelist containing files with observation data $y_i = (\mu_{\psi_i}, \sigma^2_{\psi_i})$ indexed by location x_i, input variables from Table 2.
Result: Optimized β parameters for mean function and θ parameters for covariance kernel, gridded Gaussian process marginal means and variances or a sample from the Gaussian process evaluated in a grid.

```
1  Initialization: Create grid according to area and ω, define k(x, x′) and m(x, t), initialize state;
2  if learn_m = 1 or learn_k = 2 then
3      for file in filelist do
4          D ← ReadData (file);
5          for (x_i, y_i) ∈ D do
6              if Bernoulli(η^i_train) then
7                  AddToState(state, x_i, y_i);
8          end
9      end
10 end
11 if learn_m then FindLocalMeanfunCoeffs (state);
12 if learn_k = 1 then
13     ReInitializeState (state, fulldomain);
14     for i ← 1 to n_synthetic do
15         (x_i, y_i) ← SampleFromPrior ( );
16         AddToState(state, x_i, y_i);
17     end
18 end
19 if learn_k ≠ 0 then
20     FindCovfunCoeffs (n_ref)
21 end
22 if not sampling then
23     (n_sd, (sd_i)_{i=1}^{n_sd}) ← Decompose(n^max_dom, area, ω);
24 else
25     assert (n_gp < n^max_dom);
26 end
27 if sampling then for i ← 1 to n_sd do
28     ReInitializeState (state, sd_i);
29     AddSubdomainData (state, filelist, sd_i, η_sample);
30     for x* ∈ sd_i do
31         A*^obs ← SelectObservations(state, x*);
32         μ*, σ*² ← ComputeMarginal(x*, A*^obs);
33         if sampling = 2 then
34             ψ̂* ← Normal(μ*, σ*²);
35             AddToState(state, x*, (ψ̂*, σ²_synthetic))
36         end
37     end
38 end
```

Figure 5. Overview of satGP execution. After initialization, data are read for training m and k, and possible MRF computation is carried out. This is followed by sampling the prior if a synthetic study is performed and learning the $\boldsymbol{\theta}$ parameters controlling k. Gaussian process marginals are then computed in a grid, potentially by decomposing the domain for large grids. Finally, samples from the GP may be drawn. The names of the subprograms here deviate from those in the code to improve readability.

For constructing the set of potential observations, the grid is searched for locations that may have informative observations for the current test input stored in the state object. These locations are first ordered into categories with decreasing potential covariance, and for the best locations, which together hold at least 2κ observations, the covariance function with the test input is evaluated. Out of these, the κ best are chosen. The factor of 2 can be increased for the wind-informed kernel, and the value 8 is used in the demonstration in Sect. 4.8.

The function `ComputeMarginal()` constructs the covariance matrix $\mathbf{K}$, inverts via the Cholesky decomposition, and solves Eqs. (9) and (10) to find the marginal distribution at any test input $\boldsymbol{x}^*$. That function returns the negative log-likelihood and is therefore directly used in learning the covariance parameters $\boldsymbol{\theta}$ in `FindCovfunCoeffs()` on line 18.

The Gaussian process algorithm is an interpolation algorithm when observation noise is 0, and interpolation algorithms may misbehave when used for extrapolation. In a spatiotemporal large grid, when sampling $= 2$, i.e., when draws of the Gaussian process are generated in a regular spatiotemporal grid, computing conditionals based on the previous predictions would amount to extrapolation if done in order. For this reason, a deterministic sparse ordering is used, which ensures that test inputs corresponding to simultaneous predictions are far from each other so that their mutual covariance is negligible. Conditioning on already computed values is therefore for the vast majority of GP evaluations interpolation instead of extrapolation.

4 Results and discussion

In this section we present several simulation studies. The first experiment examines parameter identifiability with the multi-scale kernel using satGP-generated data. We then demonstrate how satGP posterior distributions look like compared to truth using synthetic ozone fields from the WACCM4 model.

After that we concentrate on analyzing satGP results produced using the OCO-2 Level 2 data. First, we learn the parameters of the locally varying mean function of the form in Eq. (2) by computing the MRF, and those fields are then analyzed. We then learn the covariance parameters of the OCO-2 XCO_2 spatiotemporal field from data. Knowing both the mean and the covariance functions allows us to evaluate the Gaussian process globally in a grid, and we present snapshots of the global mean and uncertainty fields. The section concludes by comparing posterior marginal fields generated by using single-scale and multi-scale kernels and by demonstrating how the wind-informed kernel works.

4.1 Parameter identifiability with the multi-scale kernel

We performed a synthetic study to confirm the identifiability of the multi-scale covariance function parameters. The synthetic data were generated by satGP by sampling from zero-mean processes with known covariance parameters and with a random spatial pattern from the prior, adding 1 % noise. The parameters were then estimated by computing the posterior mean estimates using adaptive Metropolis.

The identifiability experiment was performed with various kernels, and recovering the true parameters was the more difficult the more complex the kernel was. With a single Matérn, exponential, or periodic kernel, the parameters could be recovered very easily. This was also true for a combination of exponential and Matérn kernels with a relatively small κ parameter.

The covariance kernel parameters were still recoverable with a combination of three kernels, Matérn with $\nu = \frac{5}{2}$, exponential, and periodic. This setup required using a larger $\kappa = 256$. With small κ, some of the parameters had a tendency to end up at the lower boundary, possibly due to effects of the covariance cutoff on the determinant of the covariance matrix in Eq. (22). Optimization using minimization algorithms such as Nelder–Mead, COBYLA (Constrained Optimization BY Linear Approximation), or BOBYQA (Bound Optimization BY Quadratic Approximation) tended to often end up in local minima, and for this reason MCMC was used instead. The number of random reference points in E_{ref} in Eq. (24) was set to 12, which was enough to reliably recover parameters close to the true value.

The parameter limits, true values, and posterior means of the synthetic experiment with three kernels are given in Table 3. In total 200 000 observations were created in the region between -10 and 10° latitude and -10 and 10° longitude over a period of 4 years according to the true values reported in Table 3. A total of 10 million MCMC iterations were computed to make sure that the posterior covariance stabilized. The posterior, with first 50 % of the chain discarded as burn-in, is shown in Fig. 6

How well parameters can be learned from data depends always on the data and the exact Gaussian process form chosen. While the identifiability studies presented here show that the parameter calibration procedure works and that covariance parameters are recoverable in a synthetic settings, identifiability cannot be always expected. Still, even in these cases, the MAP and/or posterior mean estimates of the covariance parameters should provide good point estimates for $\boldsymbol{\theta}$.

4.2 Posterior predictive distribution from synthetic WACCM4 ozone data

A synthetic study using WACCM4-generated ozone data was conducted to verify and to illustrate that the methods to learn

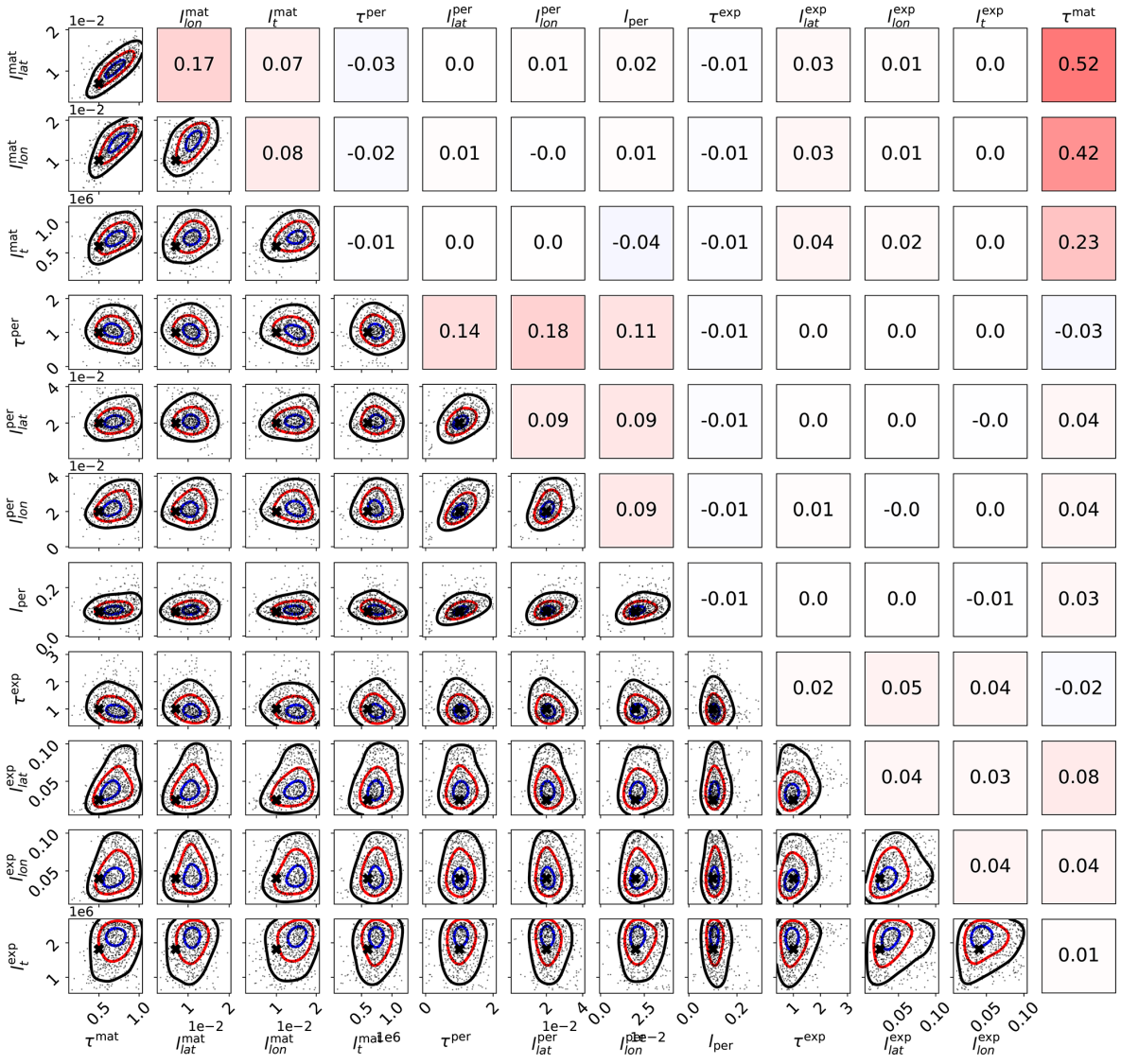

Figure 6. Scaled MCMC posteriors from a synthetic study where data were generated with a multi-scale Gaussian process. The figure demonstrates that even with three subkernels, multi-scale Gaussian process kernel parameters can be recovered. The lower-left part shows the pairwise marginal distributions of the parameters, and the black crosses denote the true parameter values. The axis labels are on the left and below the figure. The upper-right triangle shows sample correlations between the parameters from the chain, with axis labels on the left and on the top. Small within-subkernel positive correlations are present. The contours shown include 85 % (black), 50 % (red), and 15 % (blue) of the posterior mass.

the model parameters $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, and $\boldsymbol{\theta}$ produce a realistic GP regression model that then produces credible posterior predictive fields. In a synthetic setting the mean values of the posterior predictive distributions should be close to the true fields, and the discrepancies between the ground truth and the predicted fields need to be explainable by the predicted marginal uncertainties. The role of this part in the study is to give an example of how a Gaussian process predictive posterior field produced with satGP compares with the underlying true field.

The WACCM4 model is an atmospheric component of the Community Earth System Model from NCAR (Hurrell et al., 2013), capable of comprehensively representing atmospheric chemistry and modeling the atmosphere up to thermosphere. WACCM4-generated ozone data for the years 2002–2003, with a latitude–longitude grid resolution of $1.9° \times 2.5°$, 88 vertical levels going up to roughly 140 km, and an internal time step of 30 min, were used as ground truth and to generate synthetic observations. Since the model was used for generating synthetic two-dimensional data, a specific atmospheric sigma hybrid pressure level of 3.7 kPa was selected.

Ozone data at approximately 400 locations were sampled daily over a two-year period in a random pattern from the do-

Table 3. Lower and upper limits, with true and estimated parameter values. The three-kernel synthetic covariance function parameter estimation problem is already very difficult, here resulting in slight overestimation of the parameters of the smallest kernel.

	Low	High	True	Est	$\frac{\text{Est}-\text{True}}{\text{True}}$
τ^{mat}	0.05	1	0.5	0.652	0.304
$\ell^{\text{mat}}_{\text{lat}}$	0.003	0.02	0.007	0.00989	0.413
$\ell^{\text{mat}}_{\text{lon}}$	0.003	0.02	0.01	0.0135	0.350
$\ell^{\text{mat}}_{\text{t}}$	1 d	14 d	7 d	8.06 d	0.15
τ^{per}	0.01	2	1	1.073	0.073
$\ell^{\text{per}}_{\text{lat}}$	0.001	0.04	0.02	0.0207	0.035
$\ell^{\text{per}}_{\text{lon}}$	0.001	0.04	0.02	0.0220	0.1
ℓ_{per}	0.01	0.3	0.1	0.1075	0.075
τ^{exp}	0.5	3	1	0.927	−0.077
$\ell^{\text{exp}}_{\text{lat}}$	0.005	0.1	0.025	0.0352	0.408
$\ell^{\text{exp}}_{\text{lon}}$	0.005	0.1	0.04	0.0405	0.0125
$\ell^{\text{exp}}_{\text{t}}$	7 d	30 d	21 d	24.83 d	0.182

main of the experiment to learn the parameters of the mean and covariance functions. The training data set was then generated by interpolating to these points from the simulated WACCM4 data. This sampling procedure corresponds to creating on average one observation daily for each $12.5^\circ \times 12.5^\circ$ longitude–latitude square.

Using these data, the mean function parameters were fitted locally using the method in Sect. 2.3.1, utilizing the functions f_i in Eq. (11), but with two additional terms f_5 and f_6, which were similar to the f_1 and f_2 except for different Δ_{period} parameters and phase-shift parameters $\boldsymbol{\delta}$ that were shared between these f_5 and f_6 only. These functions were used to model periodical behavior with 2 and 1.5 year period lengths. The covariance function parameters of a kernel consisting of a single Matérn kernel, Eq. (15), were learned using the approximate maximum likelihood technique described in Sect. 2.6 with data from the first year. The parameter ν used for the kernel was $\frac{5}{2}$. The optimization was carried out with MCMC, and the posterior mean estimate of the covariance parameters was selected for $\widehat{\boldsymbol{\theta}}$. The values of the covariance parameters obtained were $\tau = 0.589$, $\ell_{\text{lat}} = 0.143$, $\ell_{\text{lon}} = 0.225$, and $\ell_{\text{t}} = 2$ d 16 h 15 min. That ℓ_{lon} is larger than ℓ_{lat} echoes the OCO-2 results presented later in Table 4.

For computing the posterior predictive distributions, the observational data $\boldsymbol{\psi}^{\text{obs}}$ were sampled from the WACCM4 simulations at locations closest in space and time to where the GOMOS instrument made measurements during its first year of operation. No noise was added to these observations. The posterior predictive distribution was computed for 1 full year, and the total number of observations used was 39 538. The reason for using different spatial patterns for learning the model parameters and for running the Gaussian process regression was that with this choice, the quality of the fit of the mean and covariance functions was not dependent on the spatial location, and therefore, if major spatial discrepancies between the ground truth and the posterior predictive fields had emerged, those could then have been attributed to the GOMOS sampling pattern used to generate the synthetic observations $\boldsymbol{\psi}^{\text{obs}}$.

Table 4. Covariance function parameter values learned from OCO-2 data. First column shows the Matérn kernel parameters, and the second column shows the exponential kernel parameters. The spatial length-scale parameters are given as distance on the unit sphere, with 0.01 corresponding to approximately 64 km. The length scales along the parallels, $\ell^{(\cdot)}_{\text{lon}}$, are much larger than that along the meridians, $\ell^{(\cdot)}_{\text{lat}}$.

	$(\cdot) = \text{mat}$	$(\cdot) = \text{exp}$
$\tau^{(\cdot)}$	0.899	2.72
$\ell^{(\cdot)}_{\text{lat}}$	0.00513	0.0418
$\ell^{(\cdot)}_{\text{lon}}$	0.0363	0.397
$\ell^{(\cdot)}_{\text{t}}$	20 h 22 min	16 d 20 h 12 min

The marginal posterior predictive distributions were computed globally in a uniform grid with the resolution of 2.5° in the east–west direction and 1.9° in the north–south direction between 78.63° S and 78.63° N and daily over the period from 6 January 2002 to 5 January 2003, totaling around 4.384 million marginals in the predictive posterior. The 1-year-long computation took 19 min 18 s on a relatively fast Intel i7-8850H laptop CPU.

Figure 7 shows the ground truth from WACCM4 with the mean field and corresponding marginal uncertainties obtained from satGP for 2 December 2002. The ground truth and the estimated fields are very similar, and the uncertainty is higher when there are no observations nearby. The posterior mean field retains a lot of fine detail from the ground truth and is not overly smoothed or sharp, suggesting that the covariance parameter calibration procedure has found a well-performing estimate for the covariance parameters $\boldsymbol{\theta}$. The smallest reported uncertainties are close to 0, as they should, due to lack of observation error.

4.3 The OCO-2 V9 data

The simulations with non-synthetic remote sensing data use the V9 data from the OCO-2 satellite. OCO-2 was launched in 2014, and it orbits the Earth on a Sun-synchronous orbit (Crisp et al., 2012; O'Dell et al., 2012), completing 14.57 revolutions around Earth in 1 d. The footprint area of each measurement is roughly 1.29 km × 2.25 km, but the data are very sparse in time and in space. In the presence of clouds, the satellite is not able to produce measurements, and this poses a challenge for areas with persistent cloud covers, such as northern Europe in the winter.

The present work uses the XCO_2 data, their related reported uncertainties, associated coordinate information, and zonal and meridional wind speeds that are contained in the

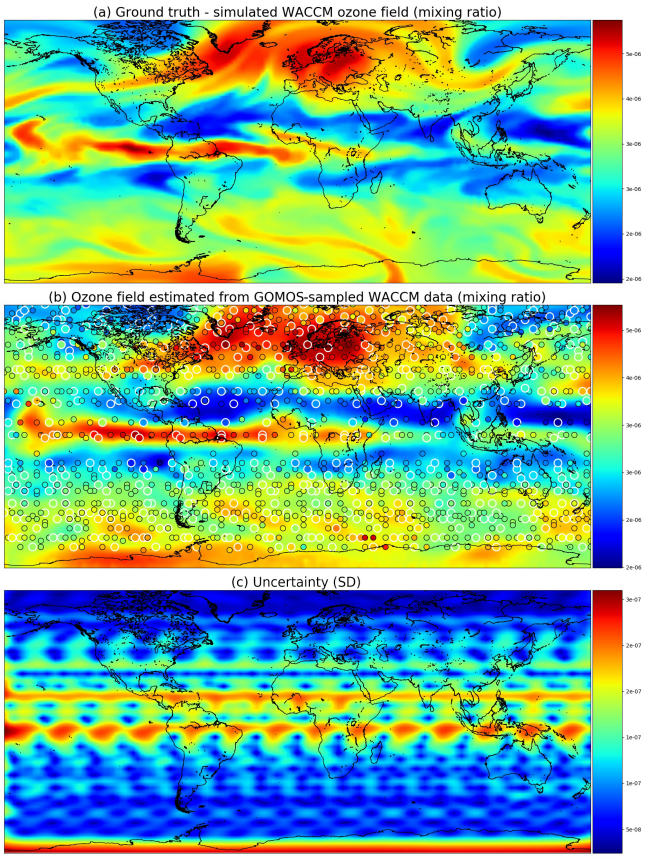

Figure 7. Ozone field mixing ratios at 3.7 kPa for 2 December 2002. Panel **(a)** shows the simulated ground truth from WACCM4, while **(b)** is the GP posterior mean, and **(c)** gives the posterior predictive uncertainties. A single Matérn kernel was used. In **(b)** the larger circles with the white edges are observations from 2 December, and the smaller circles stand for observations from 1 and 3 December.

data files. The time period considered is from 6 September 2014 to 10 November 2018, and we use only observations flagged as good, of which there are in total 116 489 342.

4.4 Solving the mean function for OCO-2 V9

Calibrating the mean function from OCO-2 V9 XCO_2 data as described in Sect. 2.3.1 produces the estimates for the $\boldsymbol{\beta}$ and δ coefficients shown in Fig. 8. The β_i parameters are the coefficients of the functions f_i in Eq. (11), and δ is the phase-shift parameter in f_1 and f_2. The upper-left quadrant of Fig. 8 shows the semiannual variability in the XCO_2 concentration. The timing of winter and summer in the Northern Hemisphere and Southern Hemisphere explains the color shift along the Equator. The lower-left quadrant shows the amplitude of the 2-times-faster oscillations, and like β_1, β_2 also shows the highest amplitude oscillations in the boreal region.

The constant term β_3 in the upper-right quadrant shows the background concentration. Some of the reddest areas such as East China, both coasts of the United States, central Europe, and the Persian Gulf stand out, and they are also areas where major emission sources are known to exist. Finding local elevated concentrations compared to surrounding areas echoes the observations made by Hakkarainen et al. (2016), where empirically defined time-integrated local XCO_2 anomalies were interpreted as possible emission sources. The trend component β_4 varies only a little spatially, due to CO_2 mixing in the atmosphere over time, and for this reason it is not shown here. The phase-shift parameter δ is modeled separately, and the field in the lower-right quadrant is obtained by optimization, conditioning on the $\boldsymbol{\beta}$ factors. This partly explains the different spatial pattern. Figure 8 shows how the phases of the annual XCO_2 cycles differ between regions, but the δ values need to be interpreted together with the β_1 and β_2 coefficients.

At high latitudes XCO_2 observations from OCO-2 are available only for a short period every year, and the quality of these measurements is often poorer. For this reason the calibration procedure may yield unrealistic and noisy values close to the poles, especially for parameters β_1 and β_2. The obtained parameter values closer than 20° to the northern and southern edges of the domain were averaged by setting the parameter values at each $\boldsymbol{x}^{ij}$ to $\boldsymbol{\beta}^{ij} \leftarrow \frac{\hat{\boldsymbol{\beta}}^{ij} d}{20} + \frac{(20-d)\overline{\boldsymbol{\beta}}}{20}$, where d is the distance to the edge of the domain in degrees, $\hat{\boldsymbol{\beta}}^{ij}$ is the calibrated parameter vector at $\boldsymbol{x}^{ij}$, and $\overline{\boldsymbol{\beta}}$ is the average value of the parameters over the area where $\boldsymbol{x}^{ij}$ is located and where averaging is performed. The δ parameter was treated similarly. This adjustment was done as a postprocessing step after finding the mean function coefficients. The main benefit of performing this adjustment is that the posterior predictive distributions become more realistic in winter at high latitudes when the mean function dominates.

Figure 1 shows time series of the mean function for a variety of locations, verifying that the exact form chosen is able to describe much of the local variability in XCO_2.

4.5 Covariance parameters of the OCO-2 V9 data

The OCO-2 data have several natural spatial and temporal length scales. The distance between adjacent observations is only 1 to 2 km in space and some hundredths of a second in time, but the distance between consecutive orbits is thousands of kilometers in space and several hours in time. On consecutive days the satellite passes close to the trajectory of the previous day at a distance of tens to 300 km depending on the latitude. The Earth has natural temporal diurnal and annual cycles, but since OCO-2 is Sun-synchronous, only the latter matters with OCO-2 data. Since the annual cycle is already fitted by finding the mean function coefficients β_1 and β_2 in Eq. (11) corresponding to the periodic functions f_1 and f_2, a periodic covariance kernel component is not included. The OCO-2 data are therefore modeled with a kernel consist-

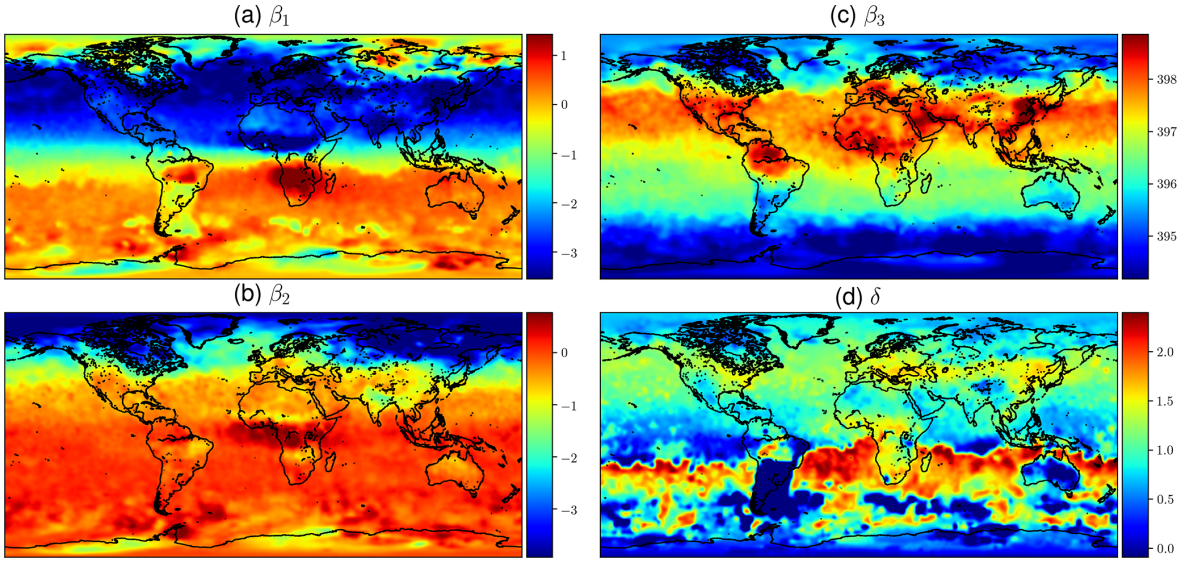

Figure 8. Mean values of mean function coefficients that were described as a Markov random field, calculated in a $2° \times 2°$ grid between 85° N and 85° S. The $\boldsymbol{\beta}_i$ coefficients multiply the f_i functions in Eq. (11). Panel **(d)** shows how the phase parameter $\boldsymbol{\delta}$ can vary more in the Southern Hemisphere where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are small. The mean function and fitted daily means for several locations with the corresponding mean function parameters are shown in Fig. 1.

ing of a larger-scale exponential and a smaller-scale Matérn subkernel.

The covariance parameters for the two-component kernel are given in Table 4. The values used were the median values from sampling the posterior with MCMC. When learning the parameters from a data set with several natural length scales, the posterior may appear multimodal, with some of the modes only having relatively little mass. In such a case, the median provides a more robust estimate for the parameters than the mean. The $\ell_{\text{lon}}^{(\cdot)}$ and $\ell_{\text{t}}^{(\cdot)}$ parameters of the posterior mean were slightly larger, which would result in slower computation. Selecting the median is further justified by the slight overestimation of some parameters in the synthetic study in Sect. 4.1.

Learning the covariance parameters from OCO-2 V9 data used the following configuration parameters for satGP: $\zeta_{\text{train}} = 0$, $\kappa = 256$, and $n_{\text{ref}} = 12$. A total of 1.1184 million MCMC iterations were completed, with the first 50 % discarded as burn-in to produce statistics. The reference points were randomly picked from a rectangle with corners at (0° S, 65° E) and (60° N, 145° E). While using the whole globe would have been a principled choice, MCMC requires lots of iterations, and for any claim of global coverage, n_{ref} would have needed to be much larger.

4.6 Posterior predictive distributions of XCO_2 from the OCO-2 V9 data

The marginal posterior predictive distribution at test points $\boldsymbol{x}^*$, given by Eqs. (9) and (10), were calculated globally in a 0.5° grid between 80° S and 80° N at daily time resolution. The first day of simulation was 6 September 2014, and the last day was 10 November 2018, spanning in total 1526 d. For each day, 230 400 marginals were computed, resulting in a collective 351 million inverted covariance matrices. The satGP parameters used were $\zeta_{\text{sample}} = 0$ and $\kappa = 256$, and the covariance kernel used was the one learned in Sect. 4.5, with parameters given in Table 4. The simulation wall time was 26 d on a moderately fast Intel i7-8700K CPU utilizing the available 12 CPU threads and 32 GiB memory.

Figures 9 and 10 present global fields of the mean values and marginal uncertainties, with a subset (to avoid excessive overdrawing) of observations shown as a scatter plot in the (a) panels. The (b) panels show how uncertainty is reduced with the overpass of OCO-2. This uncertainty reduction diminishes fast due to the Matérn component of the multi-scale kernel having a very short length-scale parameter in the time dimension. In Figs. 9a and 10a, the background color (mean of the Gaussian process posterior) usually matches the observations, but due to observational noise, the posterior mean is not everywhere an interpolated field.

4.7 Comparison of single- and multi-scale kernels with OCO-2 data

Data from the OCO-2 can be used to demonstrate how the multi-scale kernel formulation affects the predictive posterior distributions. Figure 11 shows posterior marginals from 15 September 2014. Figure 11a and b contain results from the multi-scale kernel described in Sect. 4.5, and the second row (c and d) shows fields from only the exponential part of the multi-scale kernel. The parameters of the multi-scale

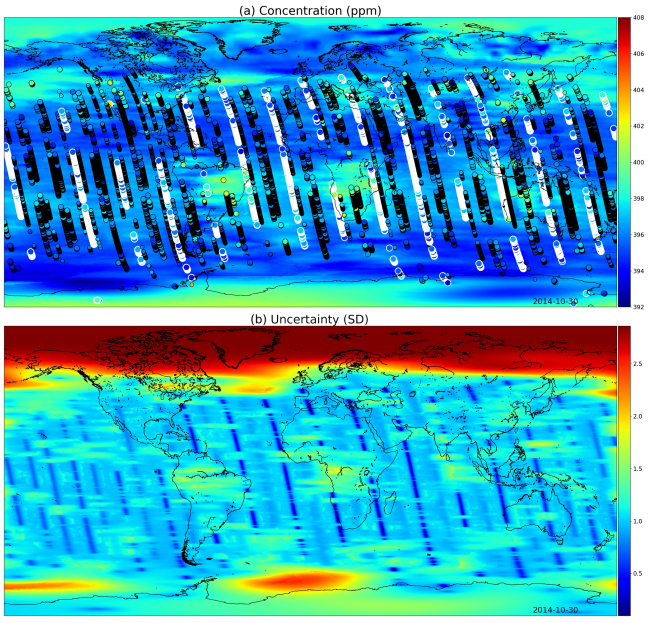

Figure 9. **(a)** XCO_2 posterior mean values and **(b)** their uncertainties on the 30 October 2014. The most informative observations are shown with the concentrations, with the large white circles being from the 30th, medium circles from 1 d before or after, and small circles from 2 days before or after. The OCO-2 utilizes sunlight for retrieval, which is why there are very few observations above 60° N. These fields include latitudes up to 85° S and 85° N.

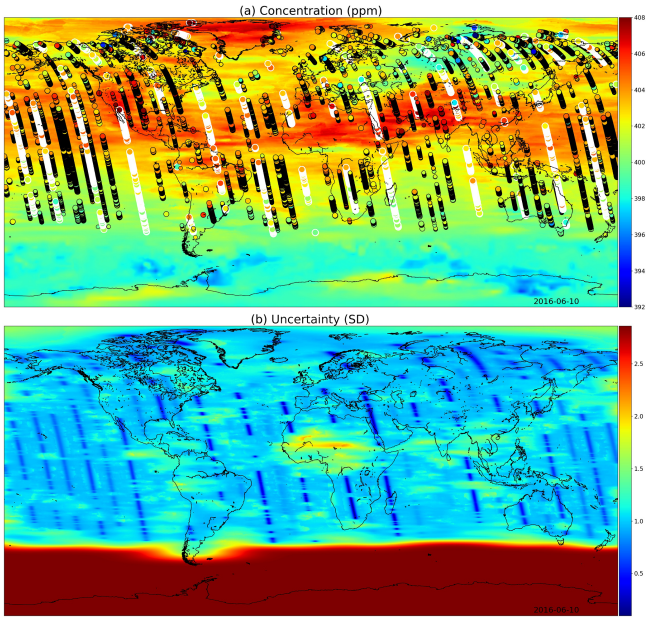

Figure 10. **(a)** XCO_2 posterior mean values and **(b)** their uncertainties on 10 June 2016. While photosynthesis in the Northern Hemisphere is already reducing the carbon dioxide concentrations globally, the observations condition the Gaussian process to higher mean values than in Fig. 9. In the summer months the uncertainty stays high close to the South Pole. These fields include latitudes up to 85° S and 85° N.

kernel are shown in Table 4. Figure 11e and f contain the difference fields between the first and the second rows. The single-kernel uncertainty is very low in Fig. 11d since lots of observations fall into regions of high covariance with almost any test input, with the exception of the northern side of Ireland, which does not have any observations nearby. Since the covariance kernel parameters were trained for the multi-scale kernel, the parameters used for the single kernel are not the ones describing the XCO_2 field best.

Figure 11a shows that as intended, the multi-scale approach leads to local enhancements of the XCO_2 mean field. Far from the measurements, the smaller Matérn kernel no longer reduces the predicted marginal uncertainties, and this leads to an increase in uncertainty in these areas. Figure 11e shows additional enhancements of the XCO_2 mean fields, which are in this case due to the different maximum covariances between the multi-scale and single-scale kernels.

The total kernel size was kept at 1024 ($\kappa = 512$ for a and b; $\kappa = 1024$ for c and d) in both experiments, and thinning and grid resolution parameter values were $\zeta_{\text{sample}} = 5$ and $\omega = 0.5°$. The very same observations were used for both simulations.

4.8 Wind-informed kernel with OCO-2 data

The wind-informed kernel, Eq. (17), lets local wind data at test input $\boldsymbol{x}^*$ rotate and scale the axes along which the covariance between two points is computed. Modeled winds are included with OCO-2 data, and they can be used to produce gridded winds that can then be used locally with the computation of each marginal posterior predictive distribution.

The covariance parameters for a single wind kernel were learned by taking the median of an MCMC posterior, similarly as was done in Sect. 4.5. The resulting parameters were $\tau = 2.07$, $\ell = 0.038$, and $\rho = 56.7$. The variance in ρ was high, possibly due to the square root in the current formulation in Eq. (19). For this simulation, $\zeta = 1$, $\kappa = 1024$, and $\omega = 0.7$, and the simulation time for the area from 27° N, 115° E to 40° N, 145° E for the single day was 2.652 s (wall time) on the i7-8750H laptop CPU.

The simulation results are shown in Fig. 12. Low uncertainties shown in blue color in (b) spread with the winds, as do the concentration estimates in (a), both due to the high reading in South Korea and the low reading close to Shanghai.

Optimally the wind-informed kernel should utilize winds that are not recomputed from the observations as was done here for convenience but rather directly from a weather or climate model or from a wind data product. The satGP pro-

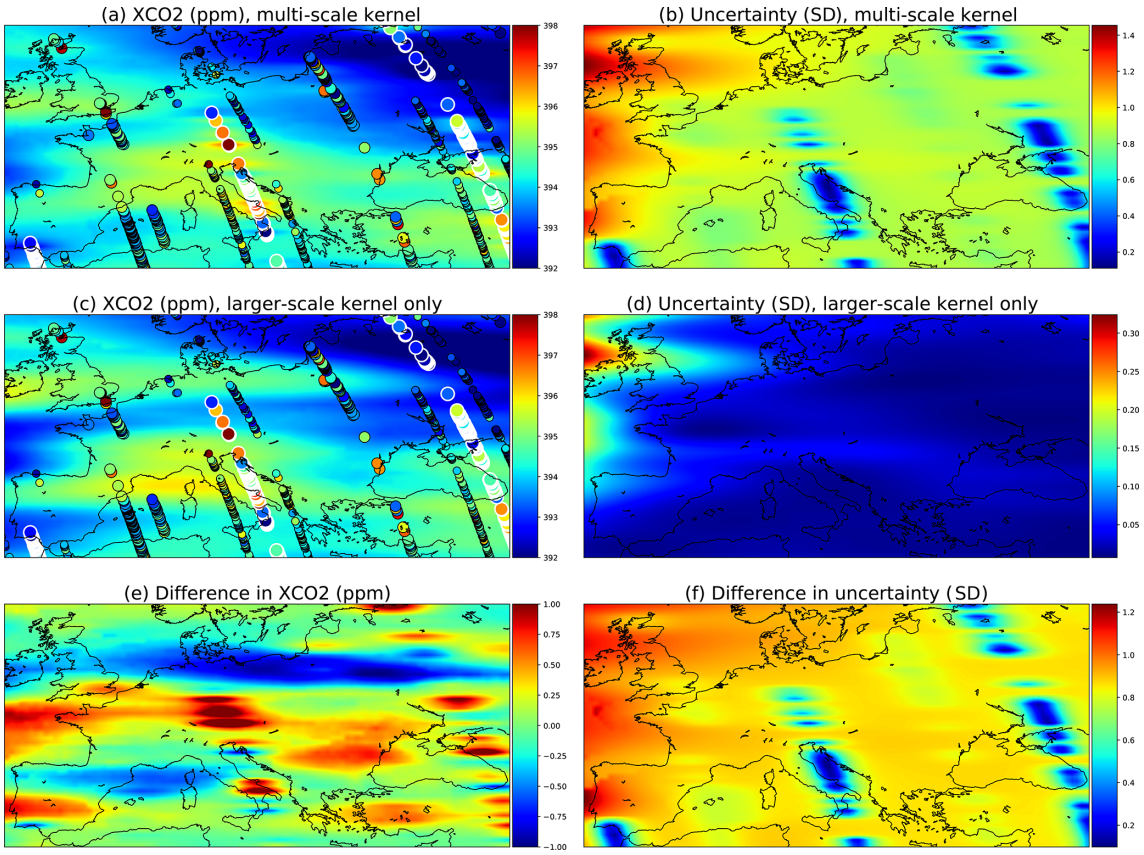

Figure 11. Comparison of a multi-scale kernel with the two components described in Sect. 4.5 and a single component kernel defined by the parameters of the exponential kernel. These parameters were given in Table 4. The observations used are the same and are shown in panels **(a)** and **(c)** as circles. The large ones with white borders are observations from the present day, 15 September 2014; medium circles are observations from the 14th and 16th, and small circles are from the 13th and 17th.

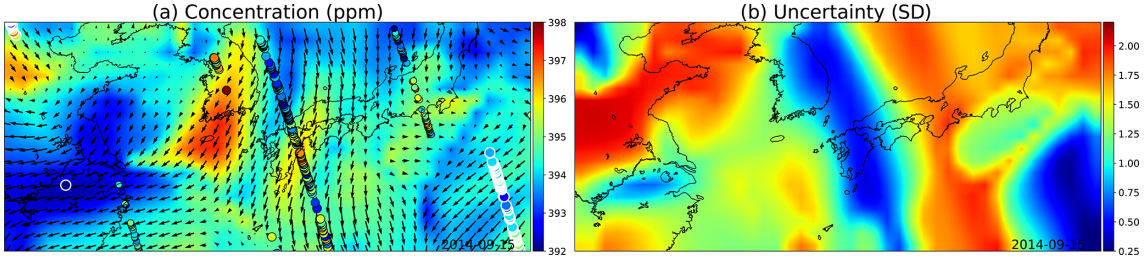

Figure 12. **(a)** GP posterior mean of XCO_2 and **(b)** its uncertainties with the wind-informed kernel. The area shown contains the Korean Peninsula in the center, China on the left, and Japan on the center right. The large circles with the white edges are present-day observations, medium circles are observations from adjacent days, and the smallest ones are observations from 2 d away. Wind direction and magnitude are given by the black arrows, and uncertainty is clearly reduced where wind is blowing directly towards or away from the observations.

gram contains configuration options for doing this. The optimal covariance function parameter values are conditional on the wind data, so the values should be learned separately for each new application and wind data set.

5 Conclusions and future work

In this work we introduced the first version of a fast general purpose Gaussian process software, satGP v0.1.2, which is in particular intended to be used with remote sensing data. We showed how the program solves spatial statistics problems of enormous sizes by using a spatially varying mean function, learned by computing marginals of an MRF, and by us-

ing a multi-scale covariance function, parameters of which are found either by using optimization algorithms or with adaptive Markov chain Monte Carlo. We also presented how satGP allows the conduction of synthetic parameter identification studies by sampling from Gaussian process prior and posterior distributions, and this could be done with any kernel prescribed, including a nonstationary wind-informed kernel. The features of satGP were demonstrated first with a small-scale synthetic ozone study and then using the enormous XCO_2 data set produced by the NASA Orbiting Carbon Observatory 2.

Various aspects of satGP can be improved in future versions, some of which include improving the observation selection/thinning scheme for statistical optimality, adding support for multivariate models and higher input dimensions, and adding methods for finding locally stationary model parameters to be able to describe heterogeneous scenes better. Despite all the room for development, satGP is a useful tool already in its present state, and it may with little additional modeling be used, e.g., to fuse data from different sources, such as GOSAT, GOSAT-2, OCO-2, TANSAT, and OCO-3. This will enable producing more precise posterior estimates, and with that a more complete picture of the evolution of for instance the atmospheric carbon dioxide distribution. Such statistically principled products that incorporate uncertainty information can then be used as a robust backbone for both making policy decisions and further scientific analyses.

Appendix A: Input parameters and variables in satGP

The satGP software by design allows for a lot of flexibility for defining how to model the quantity of interest as a Gaussian random field. This section goes over those possibilities and some practical recommendations. The parameters in Table 2 are described in more detail than earlier, along with some other configuration variables in the configuration file `config.h`. Some of the details in this section may change for future versions of the software.

Of the four sections in Table 2, the first is obvious, as those parameters control the main logic of satGP. It is recommended to first learn the mean function, then with that mean function to learn the covariance function, and only after that to calculate the means and variances for the Gaussian process with sampling $= 1$. The setting sampling $= 2$ can be used, e.g., for illustration purposes, to understanding how the different realizations of the random function would look like or to generate synthetic data products.

The area parameter defines the longitude–latitude extents of the domain where satGP performs the computations. The strings and the corresponding areas are defined in the beginning of the file `gaussian_proc.h` and can be changed there as needed. Current available areas contain, e.g., `NorthAmerica`, `Europe`, `EastAsia`, `World`, and `TESTAREA`.

The parameter n_{days} defines how many days are to be simulated after the starting day. Currently the starting day is hard coded in the code to be the first day of OCO-2 data. However, if `use_daylist` $\neq 0$ in the configuration file, a list of days can be used. This list can quite easily generated by modifying a trivial python script `create_daylist.py`, which is included with satGP.

The ω parameter determines how much spatial detail is resolved when sampling or computing marginals of the random field. A small value like 0.1 will make computing very expensive, and using such values might be unnecessary when the smallest covariance subkernel length-scale parameters are large. The spatial ℓ parameters are in the scale of distances on the unit ball, and therefore on the Equator, an ℓ parameter of 0.05 corresponds to a length scale of around 2.9°, so the ω parameter should rarely be much less than half of that. On the other hand, if the observations are spatially very close to each other and describing local variation is aimed for, then the ℓ parameters also need to be small. Given computational constraints, larger values or different area parameters may need to be used.

In the third section of Table 2, the first parameter n_{ker} denotes the number of subkernels. Even though the hard limit is set to 10, in practice this should be between one and three since the parameters of more than three subkernels are not necessarily reasonably identifiable. More kernels means more computational cost, due to the κ parameter, which is the last one in the table and is discussed later.

The parameters cfc and mf are not strictly input variables but rather C struct pointers that are created based on input variables. These variables are described in the configuration file, and they amount to choosing the covariance kernels from prescribed types (e.g. Matérn, exponential, and periodic) then defining the parameters for those kernels. The best parameters are those that are learned with learn_k $= 2$ when non-synthetic data are used.

Learning the covariance parameters $\boldsymbol{\theta}$ is best performed with MCMC, and the posterior mean and median have proven to be useful values. For unimodal posterior distributions these values are usually very close to each other. The number of MCMC iterations is controlled by the variable `mcmc_iters`, for which 10^6 is a large enough value. The number of reference points n_{ref} in the set E_{ref} in Eq. (24) that is used for computing the log-likelihood can be set to a low value of, e.g., the number of CPU threads, if at least 12 are available. If with MCMC the chain gets stuck in local minima, the value of the `mcmcs->scalefactor` in the `mcmc()` function in `mcmc.h` may be shrunk, and equally well, if the posterior ends up being flat with respect to many parameters, it may be increased. This is justified since, due to the approximate maximum likelihood method, correct scaling factor of the log posterior density is in any case unknown.

For learning the covariance parameters, parameter limits need to be given. These should correspond to the expected length scales in the data – e.g., long-range fluctuations with low-amplitude, and short-scale variations due to local effects. It is in practice best if the parameter ranges do not overlap.

If the exponent of the exponential kernel needs to be changed, that needs to be done by changing the `exponent` variable in the `covfun_dyn()` function in the file `covariance_functions.h`. Similarly, if the order of the Matérn kernel needs to be changed, that can be done by changing the variable `n` in functions `covfun_matern52()` and `initialize_covfunconfig()` in that same file.

For constructing the mean function, the configuration file contains the parameter `mftype`. The possible values are as follows: (0) a zero-mean function is used; (1) a mean function that changes only in time is used; (2) a (time-dependent) field is read in and used – this can be, e.g., the mean value from a previous Gaussian process simulation; and (3) a space- and time-dependent mean function is used. The function itself is given as a function pointer to variable `mean_function` in the configuration file, and this function needs to be defined somewhere – e.g., in the file `mean_functions.h`. For the mean function, another variable, `mfcoeff`, needs to be set. This is the total number of parameters ($\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ in Eq. 2) if `mftype` $\in \{1, 3\}$. If the mean function parameters are learned, the parameter `nnonbetas`, the number of mean function nonlinear $\boldsymbol{\delta}$ parameters, needs to be set to the appropriate value in the function `fit_beta_parameters_with_unc()` in `mean_functions.h`. For global mean function co-

efficients, the values of those coefficients are given in the configuration file, where the parameter limits for learning the space-dependent mean function parameters are also set. Finally, when learning the space-dependent mean function parameters, the smoothness of the field may be controlled by changing the `dscale` parameter in the configuration file and, to a lesser extent, by modifying the `dfmin` and `dfmax` parameters in function `fit_beta_parameters_with_unc()` in file `mean_functions.h`. Another strategy for, e.g., producing smoother mean function coefficient fields is to use high values for ζ_{train} and κ and large spatial length-scale parameters in the covariance kernel. Changing the priors for the $\boldsymbol{\beta}$ parameters is done in Sect. 2 of `fit_beta_parameters_with_unc()` in `mean_functions.h`.

In the last section, the ζ_{train} parameter controls data thinning when learning covariance kernel parameters, and the ζ_{sample} does the same when sampling $\neq 0$. How the thinning takes place was explained in the context of Eq. (26). While with few observations no thinning needs to be done at all – i.e., $\zeta_{\cdot}$ may be set to zero – with large data sets the representability of data may be improved when a coarse grid is used for computation, and also memory bottlenecks may be avoided. These parameters may be increased if faster execution is required, for example for debugging purposes.

The σ^2_{min} parameter controls which observations are not considered at all when computing at a location $\boldsymbol{x}^*$, as described by Eq. (21). The higher this is, the more data are discarded. Setting σ^2_{min} to a very low value makes searching for candidate observations slow, while picking too high a value may make posterior fields look edgy. In practice values between 10^{-7} and 10^{-3} seem to work well. This parameter is not meant to be changed often; due to this, it is set in `create_config()` in the file `gaussian_proc.h`.

The variable $n_{\text{synthetic}}$ defines how many synthetic observations are generated when learn_k $= 1$. Very large values are once again expensive, and instead a smaller area should rather be used with more moderate values of $n_{\text{synthetic}}$. Those values can be in practice up to 10^5 or more. With very low values, it may be that spatial patterns specified by the prescribed covariance kernel are not represented appropriately, and therefore values less than 10^4 should be avoided, except for maybe in setups with only a single subkernel. If $\sigma^2_{\text{synthetic}}$ is high, parameter identifiability suffers. What values are enough large also depends on the maximum covariance parameters of the Gaussian process, given by the τ^2 parameters in the formulas of Sect. 2.4.

The last parameter in Table 2, κ, defines the maximum subkernel size. The larger this parameter is, the more data are included for constructing the covariance matrix $\mathbf{K}$, whose Cholesky decomposition needs to be computed to solve the local regression problem inherent to Gaussian processes. In practice the full kernel size should be kept under 1000, and in order to compute GP calculations fast, a full kernel size of less than 500 is recommended. However, with a very small number of marginals, values up to 10^4 may be experimented with. When $n_{\text{ker}}\kappa < 64$, the speedup due to solving the GP formulas faster decreases, since at that point computing Cholesky decompositions no longer takes up the majority of the computing time. This lower bound depends on the CPU architecture and the sizes of the various CPU caches.

Whether the observations for computing the local values are chosen at random or greedily is determined by the variable `select_closest` in function `pick_observations()` in file `covariance_functions.h`. The value used should normally be nonzero, since with random selection adjacent grid points often do not utilize the best available observations closest by, leading to noisiness or graininess in the posterior mean field.

In addition to the parameters and variables listed here, there are also other parameters in the configuration file and in the code, even though those should not need to be changed. Any variables that the user might want to tweak are generally accompanied by at least some comments describing their effects.

In the current version, the satGP program is run with the script `gproc.sh`, whose comments describe the various options. Compiling and running require a modern GNU C compiler version (such as version 8) and the meson build system, and additionally all the needed libraries listed in Sect. 3. The current low version number reflects the fact that, as of now, installing and using the software will require a degree of technical knowledge, including some Python, C, and BASH programming skills.

Code and data availability. The satGP code is available as a Supplement to this paper under the MIT license. The OCO-2 V9 data used is freely available directly from NASA. The WACCM4 model is available from UCAR as a component of the Community Earth System Model.

Supplement. The supplement related to this article is available online at: https://doi.org/10.5194/gmd-13-3439-2020-supplement.

Author contributions. JS, AS, HH, and YM designed the study. TH produced the WACCM4-specific results. JS prepared this paper, wrote the satGP code, chose, tested, and implemented the computational methods, and performed the non-WACCM4 simulations, with contributions from all coauthors.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. We would like to thank Pekka Verronen and Monika Andersson from the Finnish Meteorological Institute for providing the WACCM4 data fields. The research was partly carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D0004). US Government support acknowledged.

Financial support. This work was supported by the Centre of Excellence of Inverse Modelling and Imaging (CoE), Academy of Finland, decision number 312122.

Review statement. This paper was edited by Klaus Gierens and reviewed by two anonymous referees.

References

Ambikasaran, S., Foreman-Mackey, D., Greengard, L., Hogg, D. W., and O'Neil, M.: Fast Direct Methods for Gaussian Processes, IEEE T. Pattern Anal., 38, 252–265, https://doi.org/10.1109/TPAMI.2015.2448083, 2016.

Bertaux, J., Hauchecorne, A., Dalaudier, F., Cot, C., Kyrölä, E., Fussen, D., Tamminen, J., Leppelmeier, G., Sofieva, V., Hassinen, S., Fanton d'Andon, O., Barrot, G., Mangin, A., Theodore, B., Guirlet, M., Korablev, O., Snoeij, P., Koopman, R., and Fraisse, R.: First results on GOMOS/ENVISAT, Adv. Space Res., 33, 1029–1035, https://doi.org/10.1016/j.asr.2003.09.037, 2004.

Bertaux, J. L., Kyrölä, E., Fussen, D., Hauchecorne, A., Dalaudier, F., Sofieva, V., Tamminen, J., Vanhellemont, F., Fanton d'Andon, O., Barrot, G., Mangin, A., Blanot, L., Lebrun, J. C., Pérot, K., Fehr, T., Saavedra, L., Leppelmeier, G. W., and Fraisse, R.: Global ozone monitoring by occultation of stars: an overview of GOMOS measurements on ENVISAT, Atmos. Chem. Phys., 10, 12091–12148, https://doi.org/10.5194/acp-10-12091-2010, 2010.

Chiles, J.-P. and Delfiner, P.: Geostatistics, Wiley, 2012.

Cressie, N.: Mission CO_2ntrol: 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.

Cressie, N. and Wikle, C.: Statistics for Spatio-Temporal Data, Wiley, 2001.

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_2 retrieval algorithm – Part II: Global X_{CO_2} data characterization, Atmos. Meas. Tech., 5, 687–707, https://doi.org/10.5194/amt-5-687-2012, 2012.

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E.: 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.

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.

Gamerman, D.: Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis, 1997.

Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., and Rubin, D.: Bayesian Data Analysis, Chapman and Hall/CRC, 3rd Edn., 2013.

Haario, H., Saksman, E., and Tamminen, J.: An Adaptive Metropolis Algorithm, Bernoulli, 7, 223–242, 2001.

Hakkarainen, J., Ialongo, I., and Tamminen, J.: Direct space-based observations of anthropogenic CO_2 emission areas from OCO-2, Geophys. Res. Lett., 43, 11400–11406, https://doi.org/10.1002/2016GL070885, 2016.

Hammerling, D. M., Michalak, A. M., O'Dell, C., and Kawa, S. R.: Global CO_2 distributions over land from the Greenhouse Gases Observing Satellite (GOSAT), Geophys. Res. Lett., 39, L08804, https://doi.org/10.1029/2012GL051203, 2012.

Hammersley, J. and Clifford, P.: Markov random fields on finite graphs and lattices, unpublished manuscript, 1971.

Heaton, M. J., Datta, A., Finley, A. O., Furrer, R., Guinness, J., Guhaniyogi, R., Gerber, F., Gramacy, R. B., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D. W., Sun, F., and Zammit-Mangion, A.: A Case Study Competition Among Methods for Analyzing Large Spatial Data, J. Agr. Biol. Envir. St., 24, 398–426, https://doi.org/10.1007/s13253-018-00348-w, 2018.

Hurrell, J. W., Holland, M. M., Gent, P. R., Ghan, S., Kay, J. E., Kushner, P. J., Lamarque, J.-F., Large, W. G., Lawrence, D., Lindsay, K., Lipscomb, W. H., Long, M. C., Mahowald, N., Marsh, D. R., Neale, R. B., Rasch, P., Vavrus, S., Vertenstein, M., Bader, D., Collins, W. D., Hack, J. J., Kiehl, J., and Marshall, S.: The Community Earth System Model: A Framework

for Collaborative Research, B. Am. Meteorol. Soc., 94, 1339–1360, https://doi.org/10.1175/BAMS-D-12-00121.1, 2013.

IPCC: Summary for Policymakers, book section SPM, Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 1–30, https://doi.org/10.1017/CBO9781107415324.004, 2013.

Johnson, S. G.: The NLopt nonlinear-optimization package, available at: http://github.com/stevengj/nlopt (last access: 28 July 2020), 2014.

Katzfuss, M., Guinness, J., and Gong, W.: Vecchia approximations of Gaussian-process predictions, arXiv [e-prints], arXiv:1805.03309, 2018.

Kyrölä, E., Tamminen, J., Leppelmeier, G., Sofieva, V., Hassinen, S., Bertaux, J., Hauchecorne, A., Dalaudier, F., Cot, C., Korablev, O., [Fanton d'Andon], O., Barrot, G., Mangin, A., Théodore, B., Guirlet, M., Etanchaud, F., Snoeij, P., Koopman, R., Saavedra, L., Fraisse, R., Fussen, D., and Vanhellemont, F.: GOMOS on Envisat: an overview, Adv. Space Res., 33, 1020–1028, https://doi.org/10.1016/S0273-1177(03)00590-8, 2004.

Lauritzen, S.: Graphical Models, Oxford Statistical Science Series, Clarendon Press, 1996.

Lindgren, F., Rue, H., and Lindström, J.: An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach, J. Roy. Stat. Soc. B, 73, 423–498, https://doi.org/10.1111/j.1467-9868.2011.00777.x, 2011.

Ma, P. and Kang, E. L.: A Fused Gaussian Process Model for Very Large Spatial Data, J. Comput. Graph. Stat., https://doi.org/10.1080/10618600.2019.1704293, online first, 2020.

Marsh, D. R., Mills, M. J., Kinnison, D. E., Lamarque, J.-F., Calvo, N., and Polvani, L. M.: Climate Change from 1850 to 2005 Simulated in CESM1(WACCM), J. Climate, 26, 7372–7391, https://doi.org/10.1175/JCLI-D-12-00558.1, 2013.

Nassar, R., Hill, T. G., McLinden, C. A., Wunch, D., Jones, D. B. A., and Crisp, D.: Quantifying CO_2 Emissions From Individual Power Plants From Space, Geophys. Res. Lett., 44, 10,045–10,053, https://doi.org/10.1002/2017GL074702, 2017.

Neal, R. M.: MCMC using Hamiltonian dynamics, in: Handbook of Markov Chain Monte Carlo, edited by Brooks, S., Gelman, A., Jones, G., and Meng, X., Chapman & Hall/CRC Handbooks of Modern Statistical Methods, CRC Press, 2011.

Nguyen, H., Katzfuss, M., Cressie, N., and Braverman, A.: Spatio-Temporal Data Fusion for Very Large Remote Sensing Datasets, Technometrics, 56, 174–185, https://doi.org/10.1080/00401706.2013.831774, 2014.

Nocedal, J.: Updating Quasi-Newton Matrices With Limited Storage, Math. Comput., 35, 773–782, 1980.

O'Dell, C. W., Connor, B., Bösch, H., O'Brien, D., Frankenberg, C., Castano, R., Christi, M., Crisp, D., Eldering, A., 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.: Corrigendum to "The ACOS CO_2 retrieval algorithm – Part 1: Description and validation against synthetic observations" published in Atmos. Meas. Tech., 5, 99–121, 2012, Atmos. Meas. Tech., 5, 193–193, https://doi.org/10.5194/amt-5-193-2012, 2012.

Rasmussen, C. and Williams, C.: Gaussian Processes for Machine Learning, MIT Press, available at: http://www.gaussianprocess.org/gpml/chapters/ (last access: 28 July 2020), 2006.

Rodgers, C.: Inverse Methods for Atmospheric Sounding: Theory and Practice, Series on atmospheric, oceanic and planetary physics, World Scientific, 2000.

Santner, T., Williams, B., and Notz, W.: The Design and Analysis of Computer Experiments, Springer Verlag New York, 1st Edn., 2003.

Schäfer, F., Sullivan, T. J., and Owhadi, H.: Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity, arXiv [e-prints], arXiv:1706.02205, 2017.

Tadić, J. M., Qiu, X., Miller, S., and Michalak, A. M.: Spatio-temporal approach to moving window block kriging of satellite data v1.0, Geosci. Model Dev., 10, 709–720, https://doi.org/10.5194/gmd-10-709-2017, 2017.

Vecchia, A. V.: Estimation and Model Identification for Continuous Spatial Processes, J. Roy. Stat. Soc. B, 50, 297–312, 1988.

Wainwright, M. J. and Jordan, M. I.: Graphical Models, Exponential Families, and Variational Inference, Foundations and Trends in Machine Learning, 1, 1–305, https://doi.org/10.1561/2200000001, 2008.

Yi, L., Jing, W., Lu, Y., Xi, C., Zhaonan, C., Dongxu, Y., Zengshan, Y., Songyan, G., Longfei, T., Naimeng, L., and Daren, L.: TanSat Mission Achievements: from Scientific Driving to Preliminary Observations, Chinese J. Space Sci., 38, 5, https://doi.org/10.11728/cjss2018.05.627, 2018.

Yokota, T., Yoshida, Y., Eguchi, N., Ota, Y., Tanaka, T., Watanabe, H., and Maksyutov, S.: Global Concentrations of CO_2 and CH4 Retrieved from GOSAT: First Preliminary Results, SOLA, 5, 160–163, https://doi.org/10.2151/sola.2009-041, 2009.

Zammit-Mangion, A., Cressie, N., Ganesan, A. L., O'Doherty, S., and Manning, A. J.: Spatio-temporal bivariate statistical models for atmospheric trace-gas inversion, Chemometr. Intell. Lab., 149, 227–241, https://doi.org/10.1016/j.chemolab.2015.09.006, 2015.

Zammit-Mangion, A., Cressie, N., and Shumack, C.: On Statistical Approaches to Generate Level 3 Products from Satellite Remote Sensing Retrievals, Remote Sensing, 10, 155, https://doi.org/10.3390/rs10010155, 2018.

Zeng, Z., Lei, L., Guo, L., Zhang, L., and Zhang, B.: Incorporating temporal variability to improve geostatistical analysis of satellite-observed CO_2 in China, Chinese Sci. Bull., 58, 1948–1954, https://doi.org/10.1007/s11434-012-5652-7, 2013.

Zeng, Z.-C., Lei, L., Strong, K., Jones, D. B. A., Guo, L., Liu, M., Deng, F., Deutscher, N. M., Dubey, M. K., Griffith, D. W. T., Hase, F., Henderson, B., Kivi, R., Lindenmaier, R., Morino, I., Notholt, J., Ohyama, H., Petri, C., Sussmann, R., Velazco, V. A., Wennberg, P. O., and Lin, H.: Global land mapping of satellite-observed CO_2 total columns using spatio-temporal geostatistics, Int. J. Digit. Earth, 10, 426–456, https://doi.org/10.1080/17538947.2016.1156777, 2017.