gmd-2019-156: responses to reviewer comments

We thank the anonymous reviewers and the editor for carefully reading the manuscript and for providing the very valuable comments. We address the comments one by one below. The reviewer comments are pasted verbatim below in italics, and the author responses to these comments can be found immediately under the comments, starting "A:". These are followed by "**Changes to manuscript:**" sections, where the line numbers refer to the diff file unless stated otherwise. Line numbers in the "A:" sections generally refer to the old version of the manuscript. Line numbers in the "Changes to manuscript" section refer to line numbers in the diff file.

Anonymous Referee #1

This manuscript describes a model to analyze large spatio-temporal data. Although analyzing remote sensing data of enormous sizes is no double important and challenge, the manuscript fails to describe the model and its computation details and properties sufficiently or clearly. Please see below my comments that are not necessarily ordered chronologically or by importance:

A: We thank the reviewer for this sincere assessment. To clarify the text and improve readability, we have restructured and rewritten large parts of the manuscript. This includes almost all of Sect. 2 (Methods), where the text has also been expanded in many places to more explicitly explain the technical details, with an emphasis on the requests made by Anonymous Referee #1. To aid the reader with the large number of different symbols in the manuscript (some of which were changed for clarity) we have added a full page list of symbols to help the reading. To illustrate the basic capabilities of satGP better, especially to those readers who are not so familiar with Gaussian process regression, we have added a short application to synthetic WACCM-generated ozone data, so that the reader can compare satGP output and uncertainties to the true underlying field, and appreciate that satGP is not a CO2-specific tool. We also fixed a few inaccuracies and minor bugs in the code, which lead to an increase in the version number of the software, from 0.1 to 0.1.2. Figures 1-2 and 8-10 were redone with this newest version of satGP.

1. *This manuscript suggest using the mean function of a particular form when analyzing OCO-2 data:* $m(x;\beta,\delta) = f(x^t;\delta(x^s))\beta(x^s)$ *This mean function is not a linear form of unknown parameters* $\{\delta(x^s),\beta(x^s)\}$*, noting that they are both dependent (i.e., varying) across locations. I find the description on how to estimate* $\delta(x^s)$ *and* $\beta(x^s)$ *extremely confusing.*

- *In Lines 10-20 of Page 6, it states that* β *will be estimated using the formula of generalized least squared as given in Equation (6), and* δ *will be calibrated, but no explanation is given on how* δ *will be calibrated. In addition, the authors did not explain the dimension of the matrices* F *and* K *in Equation (6). Are they large so that* K^{-1} *or* $(F^TK^{-1}F)^{-1}$ *difficult to compute?*

A1: First, in the earlier manuscript version we mention that we find a point estimate for the δ parameters before calibrating β with generalized least squares, and that we then still one more time calibrate the δ parameters. We agree that the wording could be better, and we now clarify the alternating optimization in the sentence under (7) for the revised manuscript, adding that we use optimization algorithms for the task. We also give a reference to a later section for the full description of the procedure. Second, the reviewer is right about that the matrices are too large for direct inversion. For this reason the full size of the matrix K, and by extension the computing the dense matrices mentioned above would be prohibitively expensive. In our work the size of K is up to order of $10^8 \times 10^8$, and such matrices would not fit to any computer's memory. **Changes to manuscript:** p. 7 l. 15, p. 8 l. 17-18, p.8 l. 19-24 (and the full section 2.3.2)

- *How is $\beta(x^s)$ estimated for a location x^s? For a location x^s,test without data/observation, can we estimate $\beta(x^s, test)$ and how?*

A2: The β^s is estimated via the Markov Random Field, by fitting the parameters to match the mean function to local observations, and by conditioning on the parameter values at neighboring spatial locations. When there is no data nearby, the values of the parameters will be determined by prior values (if any – we use a flat prior) and the parameters at neighboring nodes in the MRF. We agree that the description in Section 2.4 is at the moment not very clear, and we will describe the calibration procedure more clearly in the revised version. **Changes to manuscript:** We have added section 2.3.2 detailing learning β and δ for a given location, diff p. 13, last line – p. 15 l. 7.

- *Although the authors have included Section 2.4 on learning $\beta(x^s)$ as a Markov random field, this section is not connected to other parts of the manuscript but only adds confusion. It is unclear what the authors meant by modeling $\beta(x^s)$ as a Marko random field. Does this mean that the authors no longer use Equation (6) to estimate $\beta(x^s)$? What are the assumptions of this Markov random field (MRF)? What are the parameters in this MRK and how is this MRK fitted?*

A3: (Line numbers here refer to the old version of the manuscript) The β parameters are still computed with equations (6) and (7), but in addition to just computing a mean field approximation, we condition each vertex by the neighbors. This also imposes some smoothness on the posterior field of the β parameters and regularizes the problem. The fitting procedure was actually described on p. 8 l. 6-11 and in the caption of figure 2. Additionally, the conditioning on the neighbors was briefly explained in the text around p. 7 l. 27 - p. 8 l. 2. However, we agree that this description could be made clearer, and for this reason we have rewritten section 2.4 adding a lot of previously missing detail. Regarding the parameters of the MRF, the MRF is over the β parameters, and for the δ parameters we only obtain point estimates by fitting the parameters before and after obtaining the local β values (amounting to a very short alternating optimization of β and δ). The smoothness of the fitting is controlled by the `dscale` parameter mentioned on p. 26 l. 12-15, and of course also by the covariance kernel used, which affects the observation selection. The MRF is fitted according to the procedure described in the caption of figure 2. We realize that even though how the fitting is exactly done is not so critical for how the *a posteriori* Gaussian process fields look like, this procedure should be more carefully explained, and not in a figure caption. We will integrate the description in the rewritten section 2.4. (now 2.3) **Changes to manuscript:** The motivation behind the Markov Random Field paradigm is now explained in a separate subsection, 2.3.1, and learning the pointwise estimates, along with conditioning on neigbors, is now explained in the new section 2.3.2. The assumptions of the MRF are discussed first on p.12 l.9-10 and then on p.12 l.20-24. Spatial order of learning the graph is now explained on p.13 l.1-4, and elsewhere in that section.

- *It is also confusing how the parameters $\delta(x^s)$ are estimated.*

A4: The fitting of the δ parameters is carried out by optimizing them when computing the MRF as was explained on p. 8 l. 12-18. While we think that the procedure was described in the text, it could have been worded better, and we will do our best to also clarify this part of the text. **Changes to manuscript:** The δ parameter fitting has been included in the new section 2.3.2, particularly in the procedure p. 14 l.11 - p.15 l.3.

- *Line 14 of Page 8: " . . . finding $\hat{\beta}$ with Eq. (9) and (10), ..." Is this a typo? Should it be Eq. (6) and (7)?*

A5: Yes, this is a typo, this has been fixed.

- *Page 8 Line 15: The objective function $\sum_{j=1}^{n}(m(x_\nu;\beta_\nu,\delta_\nu)-\psi_j)^2+\sum_{j'\in\partial\nu}(\delta_\nu-\delta_{j'})^2$ and the optimization procedure are poorly explained. It should be noted that the mean function $m(\cdot;\cdot,\cdot)$ involved δ and β. It is very confusing how or why this function is used to estimate δ or β individually or both of them jointly, and why it should be used this way.*

A6: This part of the text describes fitting the phase-shift parameters δ, also mentioned above. For the "why" question, it is mentioned in the text that the nonlinear parameters cannot be calibrated the

same way the β parameters are dealt with. The first term blindly fits the mean function to data, while the second term imposes smoothness on the δ-field. For simplicity and speed we don't use a dense error covariance matrix for the first term (as in ordinary least squares as opposed to generalized least squares), since for the δ parameters we are not interested in uncertainties. This is a modeling choice with which we aim to satisfy two objectives: first, to get reasonable estimates of the δ field (for total column CO2 we expect that the spatial variation of the phase parameter should be be smooth) so that we do not end up fitting noise, and second, to perform this without the need to handle covariances in the optimization. While taking to account observation covariances by computing e.g. $(m(x_\nu;\beta_\nu,\delta_\nu)-\psi_j)^T K^{-1}(m(x_\nu;\beta_\nu,\delta_\nu)-\psi_j)$ instead of plain squared error in the first term would be possible, we do not think that would really improve the fit for the δ parameters: this can be verified by e.g. looking at Fig. 1, which we have updated to show the fit to the actual observations instead of the daily means. Looking at that figure, it is clear that the phase shift δ parameters are correctly estimated. For this reason we are not concerned about the effect of this compromise to the precision of our mean function. Last, we'd like to emphasize that the covariances are properly accounted for when finding the β parameters, so this compromize only affects the δ parameters.

The "how" part of the question was addressed in the comments above in A1, A3, and A4. We will still add a note about how the graph structure could be solved with algorithms such as generalized belief propagation, implementation of which is not yet included in satGP. This is future work that we hope to find time for at some point.

As a final note we'd like to point out that the form of the mean function is generally data set specific, and it is the task of the modeler to understand the mean behaviour of the field before learning the GP parameters. While other data sets may require different, perhaps more complicated, mean function formulations, it is also possible to supply the mean function to satGP directly as an array. **Changes to manuscript:** Optimization procedure is now explained much more carefully (p.14 l.11 - p.15 l.3). We also now mention the mean function in that section (p. 14 l. 5) to remind the reader of the context. Generalized belief propagation is mentioned as a possible future inference algorithm for the MRF on p.13 l.7

2. *The notations in this manuscript are very confusing overall. For example, the authors sometimes use* $\beta(^s)$ *and later use* β_ν*. The covariance parameters are even more confusing. There are* l, l_c*, and* l_I *. Even the definition of* I *is not consistent: It is originally stated* $I \subset \{x^s, x^t\}$*, but later used as* $I = STorI = S$*, and* $I = ST$ *. Also, the authors used* Δ_{year} *in Equation (11) and stated* Δ_{year} *is the duration of one year, does this mean* $\Delta_{\text{year}} = 365$*? Similarly, in Equation (15), the authors used* Δ_{period} *; is it 365 as well?*

A7: First, we agree that using both $\beta(x^s)$ and β_ν may be confusing. We use ν to refer to a generic vertex on a graph, whereas we used $\beta(x^s)$ on p. 7 l. 22 to underline that the β parameters are space-dependent. We have removed this latter notation and explain the connection of the β_ν to the spatiality of the problem better.

Second, regarding the different ℓ variables, we'll do our best to make the notation more consistent. The reviewer is correct to point out that more clarity is needed. We have made the notation more consistent and added these to a table of symbols.

Third, regarding the index set notation with the letter I, we agree that this is not optimal, and that the notation is not consistent (there is e.g. both ST and I_{ST} etc.). We have now made the notation consistent and no longer needlessly give the I variables as arguments of the covariance functions. We also explain this notation in the table of symbols.

Fourth, the Δ_{year} vs. Δ_{period} was an intentional discrepancy: we use a period length of one year for the OCO-2 data (this is a modeling choice) but for instance the now-added WACCM example uses also 1.5 and 2 year periods. We have therefore removed the Δ_{year} notation altogether. **Changes to manuscript:** We added a full-page table explaining the most often used symbols and their dimensions, p.11. We explain the I-related symbols on p.15 l.18-19,21-22, and also in the table of symbols. The ℓ-symbols are clarified, e.g. p.15 l.18,20,25-26. The notation Δ_{year} has been removed. We now use $\triangleq$ to emphasize that an equation is a definition.

3. *The authors suggest the multi-scale covariance function given in Equation (18):* $k(x,x';\theta) = \delta(x,x')\sigma_x^2 + k_{\text{per}}(x,x';\theta,I_S) + k_M(x,x';\theta) + k_{\exp}(x,x';\theta,I+ST) + k_{\text{W}}(x,x';\theta,I)$.

- *First, I am not sure multi-scale is an accurate way to describe this covariance function. I feel this function is to add different types of covariance functions together, but these components not necessarily differ in terms of scales.*

A8: It is true that the combined covariance function works by adding different covariance functions together. However, how we decided to call the combined covariance function had a lot to do with the intended use of satGP: in the OCO-2 case we are in particular interested in finding the different length scales in the data induced by both spatial sparsity and underlying processes. Furthermore, remote sensing data often describes data from processes that involve different various characteristic length scales, as presented in e.g. figure 9. While we could of course call the full kernel "multi-component", we would rather like to emphasize that we are specifically interested in the different length scales. Note, that even if the kernel components are of different types, they still may describe processes at different length scales. A non-multiscale kernel would arise in a situation, where a kernel utilizes, say, an exponential and a periodic kernel component with the same length scale parameters. Such usage, while possible, would likely be slightly unusual. For this reason we'd like to keep the terminology that we currently have. We will, however, add a note that the kernel could also be called "multi-component", and briefly explain the reasoning behind the multi-scale name. **Changes to manuscript:** We mention that multi-component could be an alternative name, p.18 l.6-9.

- *The authors did not explain clearly the component $kW(x, x0; \theta, I)$. Although Equation (16) states it is equal to $k\exp(x_W, x'_W; \theta^W, ST)$, the authors fail to explain x_W or the quantifies in Equation (17) especially, l, l^t, $l_\parallel$, and $l_\perp$, and how these parameters are chosen/estimated.*

A9: We agree that this explanation is not adequate. We now clarify how the rotated kernels function and rephrase this part of the text to improve clarity. As with other covariance kernels, also these parameters may be found by maximum likelihood. This procedure is outlined in Section 2.7, but we will add a note that it applies also to the wind-informed kernel parameters. **Changes to manuscript:** We have rewritten the section explaining the wind kernels, p.16 l.17 - p. 17 l.15, and we now explicitly give formulas for x_w and $\ell_\parallel$ and $\ell_\perp$, and explicitly list ℓ_t and ℓ in the parameters of k_w. We explain that the ρ parameters may be learned like the other parameters (p.20 l.17)

- *What will happen if there are missing data in wind velocity?*

A10: In case of OCO-2 (and with many other products), the wind data is included with the data files. The satGP code also includes running a Gaussian process for the wind data (and the output can then be utilized with k_w). Wind data may also be read from an external file. We will add a note about these capabilities in the text. **Changes to manuscript:** We now mention how wind data may be read in and that it is a required input for k_w, p.17 l.13-15.

- *Why isn't there an I involved in the Matérn component $k_M(\cdot, \cdot; \cdot)$?*

A11: Yes, there should of course be. However, we decided instead to remove the I arguments from all the kernels, since changing the dimensions over which the covariance functions work requires changing the code. **Changes to manuscript:** We have removed the I arguments from kernels in equations 14-17, p.15-16.

- *For the exponential component, the definition given in Equations (12) and (13) are not clear. At least there are two ways to define this component:*

$$k\exp(x, x_0; \theta, I_{ST}) = \tau^2 \exp\left(-\left|\frac{x - x'}{l_{ST}}\right|^\gamma\right)$$

or

$$k\exp(x, x_0; \theta, I_{ST}) = \tau^2 \exp\left(-\left|\frac{x^s - x^{s'}}{l_s}\right|^{\gamma_s}\right) \exp\left(-\left|\frac{x^t - x^{t'}}{l_t}\right|^{\gamma_t}\right)$$

dependent on whether the spatial and temporal components share the scale or exponent parameters. I don't know what the authors have used, and there is no justification of their choice.

A12: Each dimension has its own scale length parameter. This is what the subindex c in the sum and also in the term ℓ_c in (13) refers to. The sum is over the dimensions in the set I, and while we think this is quite clearly presented, we will still try to clarify. This means that the second version listed above is what is being used, with the caveat that the exponents γ are the same. If needed, this restriction can of course be quite easily lifted by modifying the code. For the OCO-2 experiments the exponent 2 was used. **Changes to manuscript:** We try to explain the notation better, p. 15 l.16-27. We underline that the dimensions are independent and have separate length scale parameters, p.15 l.24-25. We have changed the notation to contain less subscripts, e.g. $\xi^{\gamma}_{\ell_I} \Rightarrow \xi^{\gamma}_{I}$ in equation 13.

- *The authors need to provide a better description of these components in the covariance function and explain why they are identifiable based on their formulations and definitions. Also, it is necessary to clarify whether some parameters are the same or vary across these components, such as* τ^2 *,* γ*, and* l*.*

A13: We now clarify that parameters such as τ are different for each kernel component. They can be found from the data, as was shown in the OCO-2 case. Of course the reviewer is correct that parameters of an arbitrary set of kernels would not necessarily be identifiable. However, what set of kernel components are chosen, is up to the modeler and depends on the data used. In the synthetic experiments we show that length scales of even three kernel components are recoverable, even though some parameters were slightly overestimated. We did perform additional tests, according to which parameters of two-component kernels are recoverable without such overestimation. We will add a comment on the modeler's role in picking the set of kernel components, underline that the synthetic studies verify the identifiability of the parameters, and furthermore do our best to improve the description of the kernels in general. **Changes to manuscript:** We clarify that the parameters differ over the different kernel components, by subscripting the parameter vectors θ, p. 15 l.28, p.16 l.1,8,22, that γ is shared across dimensions, p.15 l. 25, and that ℓ parameters are different for each dimension (p.15 l.24-25). We state that the combined covariance parameter vector is now called θ, (p.18 l.2). We have added a note about the modeler's role in modeling the data (p.18 l.3-4). We have added a simple one-kernel synthetic example, Sect. 4.2, which shows that the techniques used for learning mean function and covariance function parameters produce very good-looking fields, and that the uncertainties are what should be expected, implying that the method for finding the covariance parameters is able to converge to a well-performing parameter estimate. (p.26 last line - p.29 l.8).

4. *I find Sections 2.6 and 2.7 quite difficult to understand. It seems that the authors use local kriging, that is, using a subset of data close to a prediction location* x^* *to estimate the covariance parameters and to make prediction.*

A14: This is correct. We use a set of hyperspheres in the space of the inputs x, within which we fit the kernel parameters. **Changes to manuscript:** We have significantly expanded and revised/rephrased/rewritten both of these sections to improve readability.

Furthermore, it appears that the authors use different subsets of data to estimate the components in the covariance function. Why not using a single subset data to estimate the entire covariance function? Or, were the authors trying to avoid identifiability issue by using different data sets to estimate different covariance components? If a subset of data are used, I assume the size of this chosen subset is not too large, but why is there a need to use a block diagonal matrix $\tilde{K}$ *as in Equation (22)? This approximation is not clearly explained, neither is* E_{ref} *in Equation (22). 2 ?*

A15: We use the same subset of data to fit all the components at once, otherwise we could hardly claim that the parameters we choose are somehow optimal or correct. The sequentiality of the observation selection is due to something different: when we choose the (one and only) set of observations for fitting covariance parameters, we need to pick them so that all the (expected) length scales are represented in the data set. For instance, if the length scales are 10 kilometers and 1000 kilometers, we need to include both local dense data, and data from further away: if for instance we only include the closest observations, we don't really have leverage to say much about the behavior over longer length scales. We would like to point out more generally, that parameter identifiability is conditional on the data, so with some data (for instance with only one or zero observations) there will always be identifiability issues. While we think that we actually do explain what E_{ref} is on p. 12 l. 24, we agree that the description is short, and that the block-diagonality is explained only implicitly (or not at all). We will clarify these points and include a better description of the $\tilde{K}$ matrices in the revised manuscript.

Changes to manuscript: We underline that we use a single data set (p. 18 l.22-23). We now clarify the block-diagonality and the relationship between K and $\widetilde{K}$ in the text (p.20 l.24-30) We also disambiguated the notation in (new) Sect. 2.6 and added a short algorithm (figure 4) to describe the observation selection. We mention that the $\mathrm{E}_{\mathrm{ref}}$ is a set of random points from the domain (p. 20 l.12-13)

Moreover, in Equation (19), should it be $> \sigma_{min}$ *rather than* $< \sigma_{min}$*?*

A16: This is definitely true and has now been fixed.

5. *The authors mentioned the nearest neighbor Gaussian process, but did not cite the reference correspondingly.*

A17: Thank you for pointing this out, we have now added a proper reference. **Changes to manuscript:** Added reference, p.4 l.9

6. *It is unclear where or why MCMC is needed and how it is implemented (prior specification etc.). The authors described optimization in Section 2 and also in the first paragraph of Page13. However, later in Page 17, the authors stated that MCMC is used instead. Section 2 does not describe MCMC.*

A18: While learning the mean function parameters β and δ utilizes optimization with the BFGS algorithm, covariance parameters are learned using MCMC. The likelihood for learning the covariance parameters is noisy due to the observations selected changing with changing parameter values. For this reason optimization algorithms tend to get stuck in local minima. This was actually mentioned on p. 17 l. 4-5 (old version of MS). We do mention that an Adaptive Metropolis implementation is included in the code, and that that can be used for finding the parameters (p. 13 l. 1-5). It is true that the priors are not described. We use flat priors, and will add information about them in the text in sections 4.1 and 4.4. We will also add a short description of MCMC to section 2.7. **Changes to manuscript:** MCMC and the motivation and its relation to optimization are now explained in more detail on (diff) p.21 l.4-17

7. *It should be Matérn covariance function, instead of Matern.*

A19: This has been fixed.

Anonymous Referee #2

A20: We thank Anonymous Referee #2 for appreciating our work. (No corrections or clarifications were requested.)

Executive Editor Comment

. . . Therefore please provide the satGP v0.1 code or provide the reasons why the code can not be made publicly available in your revised submission to GMD.

A21: We received the approval for open-sourcing satGP today from MIT, and will include the source code of the newest version as a supplement to the final manuscript version, after first adding the license headers and copyright notices.

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, FI-53851 Lappeenranta, Finland
[3]Finnish Meteorological Institute, Erik Palménin aukio 1, FI-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)

Abstract. Satellite remote sensing provides a global view to processes on Earth that has unique benefits compared to ~~measurements made~~ making measurements on the ground~~. The~~ , such as global coverage and ~~the enormous amounts of data produced come, however, with the price of~~ enormous data volume. The typical downsides are spatial and temporal gaps and ~~less than perfect~~ potentially low data quality. Meaningful statistical inference from such data requires overcoming these problems and ~~that calls for developing efficient~~ 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 ~~designed to be~~ able to handle problems of enormous sizes and ~~is able~~ to compute marginals and sample from ~~a random process with at least over hundred million observations.~~ 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.

~~The~~ We describe the mean function of the Gaussian process ~~is described~~ by approximating marginals of a Markov random field (MRF). ~~For covariance functions, Matern~~Variability around the mean is modeled with a multi-scale covariance kernel, which consist of Matérn, exponential, and periodic ~~kernels are utilized in a multi-scale kernel setting to describe the spatial heterogeneity present in data. We further~~ components. We also demonstrate how winds can be used to inform ~~the covariance kernel formulation~~covariances locally. The covariance kernel parameters are learned by calculating an approximate marginal maximum likelihood estimate~~and this is utilized to verify~~ , and the validity of both the multi-scale approach and the method used to learn the kernel parameters is verified in synthetic experiments.

~~For demonstrating the techniques above, data~~ 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~~is used~~. The satGP ~~program is released as open source software.~~ software is released under an open source license.

1 Introduction

Climate change is one of the most important ~~current~~ present-day global environmental challenges~~, to the point where it is drawing constant widespread attention even in mainstream media~~. The underlying reason is ~~the~~ anthropogenic carbon emissions~~: among the well-mixed greenhouse gases~~. According to the Intergovernmental Panel on Climate Change, carbon dioxide (CO_2) has ~~currently~~ the strongest effect on warming the planet of the well-mixed greenhouse gases, with the radiative forcing of ca. 1.68 W m~~$^{-2}$ according to the latest IPCC report~~ $^{-2}$ (IPCC, 2013).

~~The resulting global interest in atmospheric carbon along with technological advances has resulted in several CO2-measuring satellites continuously monitoring the Earth and producing~~

Several instruments orbiting the Earth produce enormous quantities of ~~data, which are processed to~~ remote sensing data, used to compute local estimates of CO2 ~~by solving a complicated inverse problem (Crisp et al., 2012). These~~ 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), ~~which has been~~ 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), ~~which was~~ launched in December 2016. GOSAT-2 was launched in October 2018~~saw the launch of GOSAT-2~~, and in May 2019 the OCO-3 instrument (Eldering et al., 2019) was taken to the International Space Station. In addition to the CO2-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 ~~computational statistics. The many steps of producing carbon flux estimates from readings produced by satellites are summarized by e.g. Cressie (2018). In this work a tool to solve one of those steps, the production of gridded level 3 data sets with uncertainties from pointwise level 2 column integrated dry air CO2 mole fraction (XCO2) data, is introduced. Even though we demonstrate the capabilities of the softwarewith OCO-2 data, the methods are not constrained by the quantity of interest observed.~~

techniques from computational statistics, such as those implemented in the satGP software, which this paper introduces.

~~The purpose of this manuscript is four-fold. First, to introduce satGP, a fast computer program that estimates Gaussian process covariance and mean function parameters from data, computes posterior marginal distributions, and samples from GP priors and posteriors conditioning on over hundred million observations in situations where several hundred million marginals need to be computed. While lots of 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.~~ 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, ~~computational methods that allow the solution of problems of such scales are introduced. Third,~~ we present formulations of

a multi-scale covariance function and ~~mean function formulations, some~~ a space-dependent mean function, types of which we have not seen used in the remote sensing community~~, are presented. In particular, the multi-scale formulation avoids excessive smoothing, allowing one to see local effects where observations become available. Fourth, these methods are demonstrated with the XCO2 data from the OCO-2 satellite~~. We also show how these functions can be efficiently learned from data.

~~Several interesting~~ Related to this work, several kriging studies have been published before in the context of ~~satellite measurements of~~ remotely sensed CO2. Zeng et al. (2013) analyzed the variability of CO2 in both space and time over China ~~producing~~ and produced monthly maps from GOSAT data with slightly over 10000 observations. Nguyen et al. (2014) used a four 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 1×1.25 degrees in space and 6 days 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.

~~A~~ More recently Zeng et al. (2017) presented a global dataset derived from GOSAT ~~was presented by Zeng et al. (2017), with the spatiotemporal~~ with the spatio-temporal resolution of three days and one degree, and this study evaluated also the temporal trend of the XCO2. The results were validated against ~~both~~ observations from the Total Carbon Column Observing Network (TCCON) and against modeling results from CarbonTracker and ~~the~~ Goddard Earth Observing System with atmospheric chemistry (GEOS-Chem). ~~This study evaluated also the temporal trend of the XCO2. Similarly Tadić et al. (2017) describe~~ Tadić et al. (2017) described a moving window block kriging algorithm to introduce time dependence into a GOSAT-based XCO2 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) present~~ – for example Zammit-Mangion et al. (2018) presented fixed rank kriging ~~(FRK)~~ results based on OCO-2 data using a 16-day moving window. ~~The results again~~ In many of these studies, the obtained CO2 fields appear very smooth.

~~An interesting approach is presented by Ma and Kang (2017), who describe a fused~~ Applications to remote sensing data have also resulted in publications more focused on methods. Ma and Kang (2017) described a "fused" Gaussian process, combining a graphical model with a Gaussian process and applying that to sea surface temperature data. ~~Another interesting approach for atmospheric trace gas inversion is presented by Zammit-Mangion et al. (2015), who simultaneously model~~ In another computationally sophisticated application, Zammit-Mangion et al. (2015) simultaneously modeled both flux fields and concentrations using a bivariate ~~spatiotemporal model , utilizing~~ spatio-temporal model with Hamiltonian Monte Carlo (Neal, 2011) for sampling the posterior. ~~However, due~~ Due to computational challenges the ~~footprint area is~~ spatial area investigated in this work was very small.

~~For overcoming~~ For Gaussian processes, various approaches have been studied to overcome the difficulties posed by large ~~numbers of data, various methods have been proposed.~~ 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~~ Katzfuss et al. (2018), and Heaton et al. (2018) presents a comparison of the performance of several recently developed

methods is given by Heaton et al. (2018), spatial statistics methods with applications to MODIS data 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 underlined discussed by Ambikasaran et al. (2016).

In this work we describe an approach to solve spatial the satGP program that solves very large spatio-temporal statistics problems with hundreds of millions of data points. We do this by combining various ideas and techniques that come close to those applied in 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 while utilizing random sampling and aggressive pre-filtering of uninformative data (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 presentation of the general Gaussian process problem is based on the one given by Santner et al. (2003) and Rasmussen and Williams (2006).

A generic space and time dependent mean function of the Gaussian process is found by solving 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). For covariance modeling, a multi-scale covariance kernel formulation is given. The validity of the multi-scale approach is established via a synthetic study. Approximate methods to learn the parameters of both the covariance kernel and the mean function as implemented in satGP are outlined. Additionally, a non-stationary covariance kernel formulation for utilizing wind data for computation, partly inspired by (Nassar et al., 2017), is proposed.

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 satGPare demonstrated in practice by computing , we computed global XCO2 concentrations for a duration of 1526 days at 0.5° spatial and daily temporal resolutionwith XCO2 data from OCO-2 utilizing over , amounting to calculating 350 million marginal distributions, conditioning on 116 million observations. The number of computed marginals is over 350 million. An XCO2 observations from OCO-2. Figure 9 shows an example of how these results look likeis given by Fig. 9.. We also present a non-stationary covariance kernel formulation that utilizes wind data for computation, and use that covariance function with OCO-2 data. The utility of using winds with CO2 data has been demonstrated before by e.g. Nassar et al. (2017).

The key advances of this work are the capability to compute Gaussian process predictions with enormous remote sensing data sets, a practical way of learning the multi-scale kernel parameters and mean function parameters from data , and introduction of the flexible open source software, of which this is a first released version. Describing these developments is approached from the perspective of how the various parts of computation are implemented in the current version of satGPIn 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 the Envisat satellite. 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 manuscript is organized in the following manner: Section 2 describes the methods both generally and as implemented in satGP. An overview of computation in satGPis given in Sect. Section 3 , and Sect. discusses the computational details in satGP. Section 4 presents and discusses simulation results, including a multi-scale synthetic parameter identifiability studyand two applications to , an application to synthetic WACCM4-generated data, and applications using the OCO-2 v9 dataset. 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 often 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 learnedestimated.

Intuitively, one would like With Gaussian processes, we want to learn properties of a spatio-temporal 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 problemat each desired point. This can also be crudely thought about as optimally . This is closely related to solving a spatio-temporal interpolation problem when the observations have Gaussian errors.

The underlying theory related to theory of Bayesian statistics, Gaussian processes, and Markov random fields that is used in this work is well known and therefore many of the 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 the results still remain validtries minimize the loss in accuracy. Due to the size of the problemdesire to be able to solve very large problems, some sacrifices need to be made in order to be able to obtain any solution.

This section goes through the Gaussian process formalism , and and presents both generic and the satGP-specific forms of mean and covariance functionsare described. 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 A of space-time points are multivariate normal. We denote the vector of

these points points in the spatio-temporal domain by $x \in \mathbb{R}^q$ and underline that they contain both space and time components. In this work $q = 3$, 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

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

where $m : \mathbb{R}^q \to \mathbb{R}$ and $k : \mathbb{R}^{q2} \to \mathbb{R}$ are respectively $k : \mathbb{R}^{q\times q} \to \mathbb{R}$ are the mean and covariance functions of the process parameterized by hyperparameter vectors $\beta \in \mathbb{R}^{n_\beta}$ and $\theta \in \mathbb{R}^{n_\theta}$. Note, that with these functions x and x' refer to coordinates of a single location in the spatio-temporal domain, while below it may also refer to multiple locations, depending on context.

The function m above is called the drift in kriging literature, and the expected value of the process in areas regions with no data will tend to the value of the mean functionin that areathis mean function. It is chosen to reflect the deterministic patterns in the data, and these choices the particular form picked to model m will also affect how the function k and parameters θ in Eq. (1) need to be chosenspecified. With inadequate modeling of the mean function, the obtained uncertainty estimates uncertainty estimates obtained with Gaussian process regression may end up being unnecessarily large. For instance linear trends, constant factors, 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).

The covariance function $k(x,x';\theta)$ controls the smoothness of the draws ψ from Ψ. The parameter vector θ typically contains at least one scale parameter ℓ and a parameter 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. The functions m and k are fully described in Sect. 2.2 and 2.4, respectively. Additional practical guidelines are given in Appendix A.

In what follows, the domain $\mathbb{R}^q \ni x$ is divided into two disjoint parts, one of which, $\mathcal{X}^{\mathrm{train}} \subset \mathbb{R}^q$, contains the part is the set of coordinates x_i^{obs}, where observation data (training data) was were measured, and another one, $\mathcal{X}^{\mathrm{test}} = \mathbb{R}^q \backslash \mathcal{X}^{\mathrm{train}}$, where observations were not made. Any $x \in \mathcal{X}^{\mathrm{test}}$ is below called $\mathcal{X}^{\mathrm{test}} \triangleq \mathbb{R}^q \backslash \mathcal{X}^{\mathrm{train}}$, denotes its complement. Points in $\mathcal{X}^{\mathrm{test}}$ are denoted by x^* and called *test input*inputs as is often done in the GP literature, and these points are generally denoted by x^*Gaussian process literature.

In practice marginals of the random function Ψ in Eq. (1) or samples ψ from it are evaluated (computed) only at a finite set of points. Let $\psi^{\mathrm{obs}} \in \mathbb{R}^n$ denote a vector of observations — synthetic or real —Observations generated by the Gaussian process at locations $x^{\mathrm{obs}} \in \mathbb{R}^{n\times q}$. Given a set of functions f_i for constructing the mean function, the matrix with elements $f_i(x_j;\delta(x_j^{\mathrm{s}}))$ corresponding to locations x_j with regression coefficients $\beta(x_j^{\mathrm{s}})$ is denoted by $F(x)$. For a single input, instead of $F(x)$ the notation $f : \mathbb{R}^q \to \mathbb{R}^{n_\beta}$ is used, and with that, $f(x^*) = [f_1(x^*), \ldots, f_{n_\beta}(x^*)]^T$. The joint distribution of the field at observed locations is then given by

$$\psi^{\mathrm{obs}} \sim \mathcal{N}\left(F(x^{\mathrm{obs}})\beta, K)\right),$$

~~where the covariance matrix K is defined by its elements $K_{i,j} = k(x_i^{\rm obs}, x_j^{\rm obs}; \theta)$.~~ $\{x_i^{\rm obs} : i = 1, \ldots, n\}$ are denoted by $\psi_i^{\rm obs} \in \mathbb{R}$, and the vector of all $\psi_i^{\rm obs}$ is written $\psi^{\rm obs}$. These observations may be either synthetic or real.

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

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

is used ~~, where the superindexes s and t~~ in this work. The superindexes s and t refer to the spatial and temporal parts of the generic coordinate x, ~~respectively, and $\delta(x^{\rm s})$~~ and δ are auxiliary parameters which are ~~potentially~~ space-dependent. The purpose of the ~~function $\tilde{f}$ is purely illustrative, showing that given the parameters δ, the function~~ right hand side with the function $\widetilde{f}$ is to underline that f ~~does not depend~~ depends on the spatial part of x ~~, and similarly~~ only via the space-dependent δ parameters, and that the β parameters do not depend on $x^{\rm t}$, the temporal part of x. The temporal evolution of the mean function is in this particular form determined only by the function $f(x;\delta) \triangleq [f_1(x;\delta), \ldots, f_{n_\beta}(x;\delta)]^T$, and for each f_i there is a space-dependent regression coefficient β_i. ~~This~~

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

The definition of m above is very general and can describe in practice a large number of realistic scenarios. ~~However~~Nonetheless, the form of Eq. (2) imposes the strong assumption of separation of space and time in that the β and δ 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~~ 2.2.

The covariance function $k(x, x'; \theta)$ controls the smoothness of the draws ψ from Ψ. It outputs the prior covariance of the random variables at x and x'. The parameter vector θ 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 $K \in \mathbb{R}^{n \times n}$ with elements $K_{i,j} = k(x_i^{\rm obs}, x_j^{\rm obs}; \theta)$, the joint distribution of the field at observed locations is given by

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

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.

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

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

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

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

described explicitly below in Sect.~~2.6,~~

$\hat{\theta} = \arg\min_\theta \mathcal{L}(\theta),$

~~and for the second~~ 2.6. Given a point estimate of the parameters θ and δ, the $\hat{\beta}$ parameters have a closed-form expression, ~~given a point estimate of the parameters θ and δ, is given by~~

$\mathbb{E}[\beta|\Psi^{\text{obs}}=\psi^{\text{obs}},\theta,\delta] = (F^TK^{-1}F)^{-1}F^TK^{-1}\psi^{\text{obs}}$

$\mathbb{V}[\beta|\Psi^{\text{obs}}=\psi^{\text{obs}},\theta,\delta] = (F^TK^{-1}F)^{-1}.$

$$\mathbb{E}[\beta|\Psi^{\text{obs}}=\psi^{\text{obs}},\theta,\delta] = (F^TK^{-1}F)^{-1}F^TK^{-1}\psi^{\text{obs}} \tag{6}$$

$$\text{Cov}[\beta|\Psi^{\text{obs}}=\psi^{\text{obs}},\theta,\delta] = (F^TK^{-1}F)^{-1}, \tag{7}$$

provided that the space-dependent δ and β parameters do not change between the inputs in x^{obs}. This requirement implies that the solution must be found locally. Because the matrix K here is generally a dense matrix of size $n \times n$, where n is the number of observations, and since n may be extremely large, direct inversion of this matrix is in practice impossible.

The δ parameters ~~can be found approximately by finding~~ are found approximately in this work by a three-step process: first a point estimate of parameters β and δ ~~before computing Eq.~~is computed using an optimization algorithm, second, parameters β are re-computed by Eq. (6) ~~, and by re-calibrating~~ given the estimate of δ ~~alone after~~from the first stage, and third, the δ parameters alone are re-calibrated by optimization using the newly found β 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 ~~x~~ x^* are found by conditioning on the observations. In the satGP software, the space-dependent β and δ 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 ~~points x.~~

inputs. The posterior distribution $p(\psi^*|\psi^{\rm obs},\hat{\theta},\hat{\beta})$ of the Gaussian process at ~~a finite set of test inputs~~ some test input x^* can, given point estimates $\hat{\beta}$ and $\hat{\theta}$, be modeled according to Eq. (3) with

$$\begin{pmatrix}\Psi^*\\ \Psi^{\rm obs}\end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix}f(x^*)^T\\ F\end{bmatrix}\hat{\beta}, \begin{bmatrix}K(x^*,x^*) & K(x^*,x^{\rm obs})\\ K(x^{\rm obs},x^*) & K(x^{\rm obs},x^{\rm obs})\end{bmatrix}\right), \tag{8}$$

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

$$\mu^* = f(x^*)^T\hat{\beta} + K(x^*,x^{\rm obs})K(x^{\rm obs},x^{\rm obs})^{-1}(\psi^{\rm obs} - F\hat{\beta}) \tag{9}$$

and

$$\Sigma^* = K(x^*,x^*) - K(x^*,x^{\rm obs})K(x^{\rm obs},x^{\rm obs})^{-1}K(x^{\rm obs},x^*), \tag{10}$$

and where the covariance Σ^* is the Schur complement of $K(x^*,x^*)$.

2.2 ~~Overview and objectives of satGP~~

~~The satGP program is meant to be a general purpose Gaussian process toolbox with emphasis on applicability to large remote sensing datasets.It features a selection of covariance kernels and routines for learning space-dependent mean function parameters and covariance parameters from data.With a given set of parameters, it computes posterior marginals and uncertainties at the spatial resolution desired by the user, or generates samples from the process.Drawing samples from the prior is also supported, and this can be utilized for devising synthetic data experiments to study the identifiability of the GP covariance kernel parameters. This section goes through these capabilities and relevant computational details. Since the softwere is applied in Sect. 4 to OCO-2 data, details pertaining to that particular case are included for illustration~~The formulas in Eq. (8) - Eq. 10 work equally well when the x^* contains more than one test input. However, as of now, in satGP these equations are solved for single test input at a time. When computing Ψ^* with these formulas, satGP uses observations close to x^* (see Sect. 2.5), and the values of β and δ calibrated at $x^{*\rm s}$.

2.2 Mean functions in satGP

~~The~~ Equation (2) gives the most general mean function form available in satGP~~is given by Eq. (2)~~. 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~~are available~~. The specific forms of f_i used for the OCO-2 experiments in

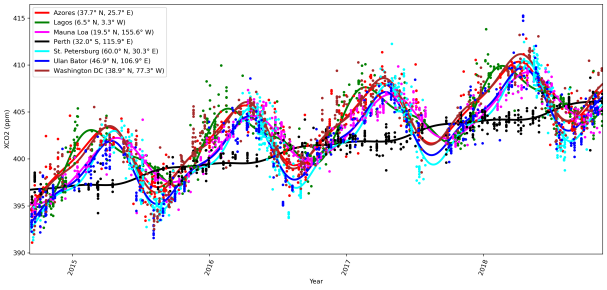

Figure 1. Mean function m with components ~~of f~~ f_i given by Eq. (11). The solid lines ~~give~~ show the mean function value for each day, fitted to ~~local data~~the XCO2 observations, ~~and~~ marked by the ~~corresponding daily means are shown as~~ dots~~of the same color~~. The ~~fit is not perfect at all times due to e.g. smoothness constraints of the field, but it works well as the Gaussian process~~ OCO-2 mean function results are discussed in Sect. 4.4.

Sect.~~4 are given by~~ 4 are

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

where ~~Δ_{year} is~~ Δ_{period} is for OCO-2 the duration of one year, and ~~$\delta_{x^{\mathrm{s}}}$~~ δ 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 ~~these~~ for any given x, f_1 and f_2 can be modeled with the same ~~$\delta_{x^{\mathrm{s}}}$~~ δ parameters. The constant term is given by f_3, and f_4 gives the slow global trend. ~~The~~ As an example of the local behavior, Fig. 1 shows the mean function fit to the ~~global~~ observed local daily mean values of XCO2 from OCO-2 ~~can be seen in Fig. 1~~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 ~~$\beta(x^{\mathrm{s}})$ as a Markov random field~~ the spatial dependence of β

When ~~not~~ satGP is not used for learning GP covariance parameters or generating synthetic training sets, the finite set of test inputs x^* for GP calculation is ~~taken in satGP to be~~ a grid with predefined geographical and temporal extents and resolution. Solving the GP marginalization and sampling problems then amounts to solving Eq. (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 ~~$\beta(x^{\mathrm{s}})$~~ β parameters as a Markov random field~~, which are often used in geophysics as a computational tool to solve large spatial statistics or inference problems. In practice what follows explains how the spatial dependence can be resolved using computational statistics~~. The MRF imposes the condition that neighboring

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 case the set that the symbol is a subset of. The domain sets in the second column are defined as $D_{\text{lat}} \triangleq [\text{lat}_{\min}, \text{lat}_{\max}]$, $D_{\text{lon}} \triangleq [\text{lon}_{\min}, \text{lon}_{\max}]$, $D_{\text{t}} \triangleq \mathbb{R}^+$, and $D \triangleq D_{\text{lat}} \times D_{\text{lon}} \times D_{\text{t}} \subset \mathbb{R}^q$, and $\mathcal{V}$ denotes the set of nodes in the graph described in Sect. 2.3.1.

Symbol	$\in$	Meaning
x	D	Generic spatio-temporal coordinate vector
x^{t}	D_{t}	Temporal part of coordinate vector x, implemented as seconds since 1970
x^{s}	$\mathbb{R}^{q-1}$	Spatial part of generic coordinate x, in practice $x^{\text{s}} = [x^{\text{lat}}, x^{\text{lon}}]^T$
x^{lat}	D_{lat}	North-south component of coordinate vector x as defined by variable area in Table 2
x^{lon}	D_{lon}	East-west component of coordinate vector x as defined by variable area in Table 2
x^{ij}	$D_{\text{lat}} \times D_{\text{lon}}$	Spatial location corresponding to i^{th} latitude and j^{th} longitude in the satGP regular grid
x^*	$\mathbb{R}^q$	Gaussian process test input – the spatio-temporal location where the GP is evaluated
x^{obs}	$\mathbb{R}^{n \times q}$	Matrix of space-time locations where the n observations in ψ^{obs} were made
β	$\mathbb{R}^{n_\beta}$	Mean function coefficients, see m below. May be space-dependent.
β_ν	$\mathbb{R}^{n_\beta}$	β coefficients for the spatial location corresponding to graph label ν in the MRF
β^{ij}	$\mathbb{R}^{n_\beta}$	β coefficients at grid point x^{ij} in the satGP latitude-longitude grid
$\beta_{\mathcal{V}}$	$\mathbb{R}^{n_\beta \times \mathcal{V}}$	β coefficients for all grid points in the satGP latitude-longitude grid
δ	$\mathbb{R}^{n_\delta}$	Space-dependent mean function parameters that cannot be learned via Eq. (6) and (7)
δ_ν	$\mathbb{R}^{n_\delta}$	δ parameters for the spatial location corresponding to graph label ν in the MRF
$\delta_{\mathcal{V}}$	$\mathbb{R}^{n_\delta \times \mathcal{V}}$	δ coefficients for all grid points in the satGP latitude-longitude grid
θ	$\mathbb{R}^{n_\theta}$	Covariance function parameters of all the subkernels of the multi-scale kernel
$\theta_{(\cdot)}$	$\mathbb{R}^{n_{\theta(\cdot)}}$	Covariance function parameters of the subkernel in the subindex $(\cdot)$
I	-	The set of all spatial/temporal indexes for each x; size of $\lvert I \rvert$ is therefore q.
I_{ST}	$\subset I$	Spatio-temporal index set: corresponding k is a function of space and time.
I_S	$\subset I$	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
$\ell_{I'}$	$\mathbb{R}^{+\lvert I' \rvert}$	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 x^{ij}
∂_ν	$\subset \mathcal{V}$	Set of nodes in the graph with edges to node ν
Ψ	-	Random field of the quantity of interest
ψ^{obs}	$\mathbb{R}^n$	Values of the observations of the field at locations x^{obs}
ψ	$\mathbb{R}^D$	Realization of the random field Ψ
$k(x, x')$	$\mathbb{R}$	Covariance function value of inputs x and x'
$m(x; \beta, \delta)$	$\mathbb{R}$	Mean function value at x with parameters β and δ, $m(x; \beta, \delta) = f(x; \delta)^T \beta$
$f(x; \delta)$	$\mathbb{R}$	Vector of functions to construct the mean function at x with parameters β and δ
F	$\mathbb{R}^{n \times n_\beta}$	Matrix with coefficients $F_{ij} = f_j(x_i^{\text{obs}; \delta})$
K	$\mathbb{R}^{n \times n}$	Covariance matrix with elements $K_{ij} = k(x_i^{\text{obs}}, x_j^{\text{obs}})$

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 MRF is~~ 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 β parameters need to be solved for each individual vertex. Point estimates of the δ parameters, mentioned in Sect. 2.1, are found at the same time with the β parameters. The intimately connected spatial and local problems are described in the subsections below.

2.3.1 Mean function parameters β 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 random variables. In satGP, an MRF is used to describe how the β coefficients depend on each other spatially. The MRF used in satGP assumes, that in addition to data, the β coefficients only depend on the coefficient values in the neighboring grid points.

Technically, the MRF in satGP is an undirected graphical model ~~$\mathcal{G} = (\mathcal{V}, \mathcal{E})$ (Lauritzen, 1996)~~$\mathcal{G} \triangleq (\mathcal{V}, \mathcal{E})$ (Lauritzen, 1996), with the set of vertices ~~$\mathcal{V} = \{\nu^{ij} | i = 1 \dots n_{\text{lat}}, j = 1 \dots n_{\text{lon}}\}$ and edges $\mathcal{E} = \{(\nu^{i,j}, \nu^{i+1,j}) | i = 1 \dots n_{\text{lat}} - 1, j = 1 \dots n_{\text{lon}}\} \cup \{(\nu^{k,l}, \nu^{k,l+1}$~~ ~~The vertices~~ or nodes $\mathcal{V} \triangleq \{\nu^{ij} | i = 1 \dots n_{\text{lat}}, j = 1 \dots n_{\text{lon}}\}$ and edges $\mathcal{E} \triangleq \{(\nu^{i,j}, \nu^{i+1,j}) | i = 1 \dots n_{\text{lat}} - 1, j = 1 \dots n_{\text{lon}}\} \cup \{(\nu^{k,l}, \nu^{k,l+1})$ We use both ν and ν^{ij} ~~correspond to the mean function parameters~~ 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 β^{ij} at grid point (i, j). ~~This Markov property~~ After finding the marginal distributions of these vectors in the graph the *maximum a posteriori* (MAP) values of β^{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 β coefficients of the nodes depend on each other. For any non-edge vertex $\nu^{i,j}$ there are edges in $\mathcal{E}$ to 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 β ~~parameters~~ parameters of latitude i and longitude j is given by ~~$p(\nu^{ij}) = \int_{\partial\nu^{ij}} p(\nu^{ij} | \partial_{\nu^{ij}}) p(\partial_{\nu^{ij}})$, where $\partial\nu^{ij} = \{\nu \in \mathcal{V} | (\nu, \nu^{ij}) \in \mathcal{E}\}$~~$p(\beta^{ij}) = \int_{\partial\nu^{ij}} p(\nu^{ij} | \partial_{\nu^{ij}}) p(\partial_{\nu^{ij}})$, where it is understood that ν^{ij} and $\partial_{\nu^{ij}}$ refer directly to the random variables, β^{ij} and the joint distribution of the β coefficients of its adjacent vertices, respectively.

~~Since the maximal cliques of this graph are the connected pairs of vertices~~The satGP program needs to compute the marginal distributions of each β^{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. ~~One reasonably efficient way to solve marginals for each vertex in such a graph is to use~~ 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 ~~uses a modified version to compute~~ 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 ~~back, conditioning each ν^{ij}~~ 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 $\partial\nu^{ij}$ ~~without introducing~~, but the diagonal edges of the ~~reconstituted graph~~so-called reconstituted graph are not introduced, as would ~~be normally done.~~

~~The program also~~ normally be done. When starting again from the bottom right corner after computing diagonals numbered $1 \ldots N$, the $(N+1)^{\text{th}}$ 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 β_ν — and any other vertex on the $(N + n_{\text{lon}} - 1)^{\text{th}}$ 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, 2 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 ~~. The structure of the MRF and the approximate elimination order are shown in Fig. 2~~by the `dscale` parameter, see Appendix A.

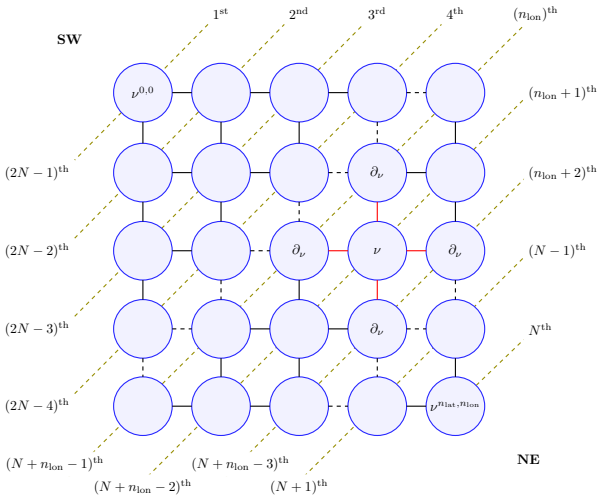

Figure 2. The marginal distribution of vertex ν, $p(\nu)$, is conditional only on the neighbors ~~$\partial_{\nu^1} \ldots \partial_{\nu^4}$~~ in ∂_ν (connected to ν with red edges) due to the Markov structure in the pictured lattice graph. ~~Each connected pair is a maximal clique in this particular case.~~ 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 ~~$N = n_{\text{lat}} + n_{\text{lon}} - 1$~~$N \triangleq n_{\text{lat}} + n_{\text{lon}} - 1$. ~~The $(N+1)^{\text{th}}$ computation~~ Southwest and northeast corners of the domain are labeled SW and NE in the ~~corner is not conditioned on already-computed neighbors~~ graph. The final values of the parameters are obtained when diagonals from N to ~~avoid double counting data~~$2N - 1$ are computed.

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

Assume that for the vertex ν in Fig. 2 the neighbors marked ∂_ν have been computed. Computing the marginal distribution of β and an estimate of δ at ν, referred to below as β_ν and δ_ν, is carried out in several steps. These steps take place inside solving the spatial problem described above: 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 δ cannot be estimated with regression ~~like~~ the way β is found in Eq. (9) and (10). For this reason, the nonlinear space-dependent δ-parameters are found with an optimization algorithm from the NLOpt package (Johnson, 2014), by default the BFGS algorithm, before finding $\hat{\beta}$ with Eq.~~(9) and (10), and after~~ (6). After obtaining $\hat{\beta}$ the δ ~~parameter is~~ parameters are re-optimized given the $\hat{\beta}$. ~~For calibrating the δ parameters for vertex~~ The full calibration process for a single graph node ν ~~, the quantity $\sum_{j=1}^{n}(m(x_\nu;\beta_\nu,\delta_\nu)-\psi_j)^2+\sum_{j'\in\partial\nu}(\delta_\nu-\delta_{j'})^2$ is minimized.Here the~~ proceeds in the following manner:

1. Select n_ν observations ψ_ν^{obs} of the observable that are close in spatial covariance to the test input x^*, in this case the spatial location corresponding to vertex ν. The selection process is described in detail in Sect. 2.5.

2. Find a best-guess δ_ν (and β_ν, which is not used) by running the BFGS optimization algorithm (Nocedal, 1980) to find an approximate *maximum a posteriori* estimate by computing

$$\widetilde{\beta}_\nu,\widetilde{\delta}_\nu = \arg\min_{\beta,\delta}\Bigg\{\sum_{j=1}^{n_\nu}(m(x_\nu;\beta_\nu,\delta_\nu)-\psi_{\nu\,j}^{\mathrm{obs}})^2 + \sum_{\nu'\in\partial\nu}\big((\delta_\nu-\delta_{\nu'})^T(\delta_\nu-\delta_{\nu'}) + (\beta_\nu-\beta_{\nu'})^T(\beta_\nu-\beta_{\nu'})\big)\Bigg\}. \tag{12}$$

   The first sum runs over the training data selected by the observation selection method described in Sect.~~2.5~~ 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 β and/or δ parameters for the individual vertices. ~~The complexity introduced by the interactions described by the edges is taken care of by~~

3. Given $\widetilde{\delta}_\nu$, an estimate of the GP covariance parameters $\widetilde{\theta}$ – e.g. from a previous simulation or a best guess – and the observations ψ_ν^{obs}, compute $\mathbb{E}[\beta_\nu|\psi_\nu^{\mathrm{obs}},\widetilde{\theta},\widetilde{\delta}_\nu]$ and $\mathrm{Cov}[\beta_\nu|\psi_\nu^{\mathrm{obs}},\widetilde{\theta},\widetilde{\delta}_\nu]$ via Eq. (6) and (7). Together these give $p(\beta_\nu|\psi_\nu^{\mathrm{obs}},\widetilde{\theta},\widetilde{\delta}_\nu)$. Since this computation used a flat prior, this is, by Bayes' theorem, proportional to the likelihood $p(\psi_\nu^{\mathrm{obs}}|\beta_\nu,\widetilde{\theta},\widetilde{\delta}_\nu)$.

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

5. Using the β_ν obtained at the previous step, re-optimize *only* the ~~approximate elimination algorithm described above~~δ_ν parameters as above in step number 2. Since β_ν is not varied, the term $(\beta_\nu - \beta_{\nu'})^T(\beta_\nu - \beta_{\nu'})$ in Eq. (12) plays no role here.

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

2.4 Covariance functions in satGP

The smoothness, amplitude, and length ~~scale~~ scales of the Gaussian process realizations are determined by the covariance kernel used~~, and this choice much determines how the result of the computation looks like~~. 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~~. For further intuition regarding the parameters, also see Appendix A.~~ , and other forms may be easily added in the code.

For convenience, let

$$\xi_{\ell_I I}{}^\gamma(x,x') \mathrel{\underset{\sim}{=}}\triangleq \sum_{c\in I}\left|\frac{x^c - x'^c}{\ell_c}\right|^\gamma = \|P^I(x) - P^I(x')\|_\Gamma^\gamma, \tag{13}$$

where $\gamma > 0$ is a parameter controlling the exponent, ~~$I \subseteq \{x^\mathrm{s}, x^\mathrm{t}\}$~~ parameters ℓ_c are length scale parameters, and I is a set of dimensions of the input~~, with x^s referring to latitude and longitude and x^t to time~~. The P^I matrix projects x onto indices/dimensions in I, ~~and~~ Γ is a diagonal covariance matrix with ~~elements ℓ_c^γ~~ diagonal elements ℓ_c^2, and the notation $\|r\|_\Gamma$ ~~means $r^T\Gamma^{-1}r$. The~~ stands for $\sqrt{r^T\Gamma^{-1}r}$, where r is an arbitrary vector of the appropriate size. For remote sensing data used in this work, space-only variables ~~are denoted I_S and~~ form the set $I_S \triangleq \{\mathrm{lat}, \mathrm{lon}\}$, and for spatial and temporal variables together ~~are denoted I_{ST}~~ the notation $I_{ST} \triangleq \{\mathrm{lat}, \mathrm{lon}, \mathrm{t}\}$ is used. Notation lat and lon refer to the spatial components of x, collectively earlier referred to as x^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 $\ell_{I'}$. All the covariance functions below depend on a parameter τ, square of which determines the maximum covariance that is attained when $x = x'$.

The exponential family of covariance functions with parameters ~~$\theta = (\gamma, l, \tau)$~~ $\theta_\mathrm{exp} \triangleq [\tau, \ell_{I_{ST}}, \gamma]^T$ is defined by the covariance function

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

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 ~~$\theta = (\nu, \ell_I, \tau)$~~ $\theta_{\mathrm{M}} \triangleq [\tau, \ell_{Isr}, \nu]^T$ is given by the covariance

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

where ~~$s = 2\sqrt{\nu}\xi^1_{\ell_I}(x, x')$~~ $s = 2\sqrt{\nu}\xi^1_{I_{sr}}(x, x')$ 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 ~~$\theta = (\tau, \ell_{\mathrm{per}}, \theta_{\mathrm{exp}})$~~ $\theta_{\mathrm{per}} \triangleq [\tau, \ell_{Is}, \ell_{\mathrm{per}}]^T$ is defined in satGP by

$$k_{\mathrm{per}}(x, x'; \theta, I_{\mathrm{per}}) = = \triangleq \tau^2 \exp\left(- \frac{2\sin^2\left(\pi \left[\frac{x^t - x^{\prime t}}{\Delta_{\mathrm{period}}}\right]\right)}{\ell^2_{\mathrm{per}}} \frac{2}{\ell^2_{\mathrm{per}}} \sin^2\left(\pi \left[\frac{x^{\mathrm{t}} - x'^{\mathrm{t}}}{\Delta_{\mathrm{period}}}\right]\right) - \xi^{\gamma}_{\ell_S} {}^2_{I_S}(x, x') \right), \tag{16}$$

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 ℓ_{Is}, and ~~the term θ_{exp} defines the parameters for the exponential functions ξ, while~~ ℓ_{per} ~~controls the periodic (inter-period) covariance length. While the periodic kernel is not utilized with the OCO-2 case studies below, it can be a useful tool in many other situations, such as with OCO-3, which due to not being on a Sun-synchronous orbit will make observations at varying local times~~determines how far the temporal covariance extends, modulo Δ_{period}.

~~An~~

satGP contains an additional covariance function ~~formulation available in satGP is one based on~~ 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 ~~the~~ surface water spread. ~~Therefore~~For this reason, if wind data is available, it is natural to ~~use it in~~ try to use it for inference with the Gaussian process.

~~The~~ We define the wind-informed covariance ~~has parameters $\theta = (\tau, \ell_I, \rho, w^*)$ and is defined by~~ kernel with parameters $\theta_{\mathrm{w}} \triangleq [\tau, \ell, \ell_{\mathrm{t}}, \rho, w^*]^T$ by

$$k_{W_{\mathrm{w}}}(x, x'; \theta, I_{\mathrm{w}}) = = \triangleq k_{\mathrm{exp}}(x_{W_{\mathrm{w}}}, x'_{W_{\mathrm{w}}}; \theta^W_{\mathcal{T}}, ST)\{\ell_\parallel, \ell_\perp, \ell_{\mathrm{t}}\}, 2). \tag{17}$$

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

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

where ~~the difference between x_W and x'_W is represented using transformed axes parallel~~ x^{s} and $x^{*\mathrm{s}}$ are the spatial components of vectors x and x^*, and where $w^{\parallel}$ and $w^{\perp}$ are the unit vectors in the lat-lon coordinates along and perpendicular to ~~the~~ wind direction at the test input x^*.

The spatial scaling ~~parameters in Eq. (14) for k_W, corresponding to the parallel to wind~~ (ℓ) parameters for k_{w}, corresponding now to the covariance scales parallel and perpendicular to ~~wind directions~~the wind direction, are given by

$$]1 + |w^*|\rho, \qquad \ell^{\perp} =_{\perp} \triangleq \ell_{,}. \tag{19}$$

~~where w^* is the wind velocity at the test input x^* and ρ scales the effect of the wind.~~ The parameter vector for the exponential kernel ~~$\theta^W = (\tau, \gamma, \ell^{\parallel}, \ell^{\perp}, \ell_t, 2)$~~then becomes $\theta_{\exp} \leftarrow [\tau, \ell_{\parallel}, \ell_{\perp}, \ell_t, 2]^T$, where the last element denotes the exponent γ used by the exponential kernel. ~~The resulting covariance ellipses~~ A number of possible covariance ellipses resulting from the transformation procedure are shown in Fig.~~3 for several wind vectors and values of ρ~~ 3. Some data sets, like OCO-2, incorporate

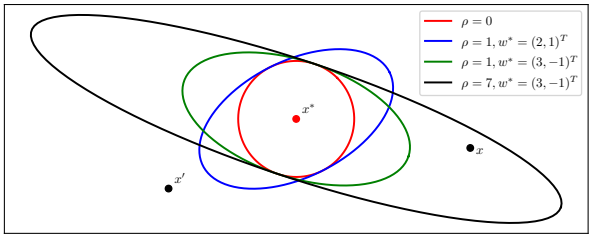

Figure 3. Equicovariance ellipses from the wind-informed kernel with various wind vectors w^* and values of ρ. The wind values are taken at the test input x^*, but the covariance function k is evaluated also for each pair of observations x and x'.

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 at is available at each 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 ~~this~~ the *multi-scale kernel* used in satGP is given by

$$k(x, x'; \theta) = \delta_{x,x'}\sigma_x^2 + \sum_{i=1}^{n_{\mathrm{ker}}} k_{\mathrm{per}}(x, x'; \theta, I_S) + k_{\mathrm{M}}(x, x'; \theta) + +k_{\exp}(x, x'; \theta, I_{ST}) + k_{W\,\mathrm{ker}_i}(x, x'; \theta, I_{\mathrm{ker}_i}), \tag{20}$$

where the first term, which in kriging is called the nugget, contains the observation error variances, and ~~the parameter θ is understood to be different for each component~~where each $\mathrm{ker}_i \in \{\exp, \mathrm{M}, \mathrm{per}, \mathrm{w}\}$.

The kernel components of a multi-scale kernel are in this work called *subkernels*. The combined set of parameters is denoted by $\theta = [\theta_{\text{ker}_1}^T, \ldots, \theta_{\text{ker}_{n_{\text{ker}}}}^T]^T$. Not all ~~kernels~~ subkernel types are included in all experiments ~~—~~ rather, the simulations in Sect. 4 utilize kernels with one to three components. ~~The kernel components of a~~ 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~~are below called *subkernels*~~.

Instead of calling $k(x, x'; \theta)$ 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 well data, 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 ~~allocation~~ selection for the multi-scale kernel

Using a large number of observations makes solving the Gaussian process Eq. (9) and (10) ~~untractable~~ 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 Eq. (9) and (10). To ~~do this, a~~ 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 ~~are defined by the user~~.

Assume that the multi-scale kernel defined by the user contains n_{ker} subkernels. For each test input x^* and for each subkernel k_l ~~let~~ the set of observations feasible for inclusion in K in Eq. ~~(8) be~~ (6) and (7) is

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

where the last condition prevents observations from being added by several subkernels. ~~From these candidate observations, $\min(|A_{*,l}^{\text{obs}}|, \kappa)$ are selected, either greedily selecting the κ observations with highest $k(x_i, x^*)$, or choosing the observations uniformly randomly sampling from those training data for which the minimum covariance threshold is exceeded, see Appendix A for additional details. When $|A_{*,l}^{\text{obs}}| < \kappa$ and $l < n_{\text{ker}}$, the parameter~~ 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 selecting more than κ ~~will be grown for the next kernel to compensate for the deficit by setting $\kappa \leftarrow \kappa + (\kappa - |A_{*,l}^{\text{obs}}|)$~~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 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 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.

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 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 $\mathcal{S}(A^{\text{obs}}_{*,i}, \kappa')$ chooses κ' observations from each $A^{\text{obs}}_{*,l}$ either greedily or randomly.

Since the ~~kernels~~ subkernels are handled sequentially, ~~the order of the different kernels may slightly~~ 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 ~~selecting all observations for all kernels~~constructing A^{obs}_{*}, the covariance matrix K is constructed by evaluating the full covariance function k according to Eq. (20) for all pairs of selected observations.

For learning the ~~locally varying parameters~~ spatially varying β and δ parameters for grid index (i, j) in the mean function with ~~Eq.(6) – (7)~~the methods in Sect. 2.3.2, the observation selection is performed by disregarding the time component on the inputs, i.e.~~setting $x_i^{\text{t}} \leftarrow x^{*\text{t}}$ for all x_i~~ by setting $x_i^{\text{obs}\,\text{t}} \leftarrow x^{ij\,\text{t}}$ for all x_i^{obs} in the training data. The reason for this is, that since learning the mean function amounts to fitting spatially varying parameter vectors β and δ, 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.

~~Observation allocation~~ Selecting the observations could be done also ~~by selecting observations~~ 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, ~~while~~ 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.

~~Out of the two methods available in satGP, random selection avoids observation sorting and is therefore faster, especially if a huge number of data are near the test input x^*. This comes at the cost of producing slightly noisier fields of marginal posterior means. For covariance parameter estimation random selection works well. The current nearest-neighbor-in-covariance approach is only one possibility, but is justified by the~~ The parameter identifiability results in Sect.~~4.1~~ 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 θ

From ~~Eq.(4),~~ Sect. 2.1 the log marginal likelihood of observations ψ^{obs} given a set of parameters θ, β and δ is given by

$$2\log p(\psi^{\text{obs}}|\beta,\delta,\theta) = -\|(\psi^{\text{obs}} - F\beta)\|_K{}^2 - \log|K| - n_{\text{obs}}\log(2\pi), \tag{22}$$

where the covariance function parameters θ ~~are implicitly in~~ implicitly determine K, and the non-linear space-dependent mean function parameters ~~in~~ δ affect the values in F~~, for which the shorthand notation $F = F(x^{\text{obs}})$ is used in this section~~. The maximum (marginal) likelihood estimate (MLE) $\hat{\theta}$ of θ can be found via minimizing

$$\mathcal{L}(\theta) = \|(\psi^{\text{obs}} - F\hat{\beta})\|_K{}^2 + \log|K| + n_{\text{obs}}\log(2\pi) \tag{23}$$

as stated in context of Eq. (5).

In the presence of a huge number of observations, calculating the determinant of the full covariance $|K|$ is not feasible, and maximizing the log likelihood is approximated ~~with the block diagonal form, resulting in~~ by

$$\hat{\theta}_{\text{MLE}} = \arg\min_{\theta} \sum_{x_i \in E_{\text{ref}}} \sum_{x_i^* \in E_{\text{ref}}} \left\{ \|(\psi_{i\text{local}_i}{}^{\text{obs}} - F_i\hat{\beta})\|_{\tilde{K}_i}{}^2_{\tilde{K}_i} + \log|\tilde{K}_i| \right\}, \tag{24}$$

where E_{ref} is a set of randomly sampled points from the ~~specified~~ spatio-temporal domain ~~.~~ specified for the experiment, determined by the parameters area and n_{days} in Table 2. The $\hat{\beta}$ and $\hat{\delta}$ parameters, the latter of which is embedded in F, are the point estimates corresponding to each 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 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 ~~vector $\psi_i^{\text{obs}} \in \mathbb{R}^{d_i}$ contains~~ vectors $\psi^{\text{obs}}_{\text{local}_i} \in \mathbb{R}^{d_i}$, where d_i is the number of observations chosen by the observation selection method of Sect. 2.5 for test input x_i^*, contain observations closest in covariance to ~~x_i, chosen according to the observation allocation rules outlined~~ x_i^*, each of which is a reference point included in E_{ref}. The matrices F_i are the corresponding F-matrices, as described in Sect.~~2.5.~~ 2.1. The last term in Eq. (23) is dropped, since while varying θ in Eq. (24) changes d_i, the ~~size of ψ^{obs} stays~~ 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 $\psi^{\text{obs}}_{\text{local}_i}$. The blocks in this approximation are the dense covariance matrices $\tilde{K}_i$, and in contrast to a full dense K, in this approximation the cross-covariances between observations in $\psi^{\text{obs}}_{\text{local}_i}$ and $\psi^{\text{obs}}_{\text{local}_j}$, $i \neq j$, are set to zero. This is done even if the randomly selected corresponding inputs x_i^* and 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 θ, 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 ~~$p(\theta|\psi^{\text{obs}},\beta,\delta)$~~ $p(\theta|\psi^{\text{obs}},\hat{\beta}_{\mathcal{V}},\hat{\delta}_{\mathcal{V}})$ is lost.

~~The covariance parameter optimization can be performed by using optimization algorithms such as COBYLA or SBPLEX available in NLOpt (Johnson, 2014). An alternative is to explore~~ By default the scaled posterior $p(\theta|\psi^{\text{obs}},\hat{\beta}_{\mathcal{V}},\hat{\delta}_{\mathcal{V}})$ is explored by using the Adaptive Metropolis (AM) Markov chain Monte Carlo (MCMC) algorithm (Haario et al., 2001), an implementation of which is included in the satGP source code. ~~Using~~ 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 $\hat{\theta} \leftarrow \mathbb{E}[\theta|\psi^{\text{obs}},\hat{\beta}_{\mathcal{V}},\hat{\delta}_{\mathcal{V}}]$ 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 ~~is just~~ simply amounts to sampling from a multivariate normal distribution which is very fast~~, and also due to that~~. 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 ~~parallellization~~ 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, LAPACKE and CBLAS, ~~are utilized and for optimization tasks , algorithms in~~ and optimization tasks are carried out with the NLOpt library~~are used~~. The computations are ~~carried out~~ performed in single precision ~~both~~ in order to save memory resources with the largest data sets~~and also in anticipation of implementing the covariance function routines in a way that allows computation on graphics processing units~~, 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 ~~and covariance kernel need to be~~ 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

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) Don't train θ, (1) generate observations and learn θ, (2) learn θ from non-synthetic data.
learn_m	`int`	0	1	(0) Don't train local β and δ, (1) find local β and δ as in Sect.~~2.3.1~~ 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 $k_1 \dots k_{n_{\text{ker}}}$ and corresponding θ, see Sect. 2.4.
mf	`struct*`	-	-	Struct pointer for defining type of $m(\cdot,\cdot)$ and associated (initial) β and δ, see Sect. 2.2.
ζ_{train}	`float`	0	∞	~~Determines what fraction of observations~~ Fraction of observations that are randomly included in ψ^{obs} when
ζ_{sample}	`float`	0	∞	~~Determines what fraction of observations~~ Fraction of observations that are randomly included in ψ^{obs} when
σ^2_{min}	`float`	0	∞	Discard observation at ~~x_i~~ x_i^{obs} for x^* if ~~$k(x_i,x^*) < \sigma^2_{\text{min}}$~~ $k(x_i^{\text{obs}},x^*) \leq \sigma^2_{\text{min}}$, see Sect. 2.5.
n_{ref}	`int`	0	∞	Number of reference points in E_{ref} in Eq. (24) for training θ
$n_{\text{synthetic}}$	`int`	0	∞	Number of random locations where synthetic data is generated for training θ
$\sigma^2_{\text{synthetic}}$	`float`	0	∞	Variance of 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.

temporal extents need to be specified and the mean function and the covariance kernel used must be given. ~~For more details than is described below, see Appendix A.~~

~~For computational efficiency, several~~ 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 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 ~~, where $r_l \propto \prod_{i=1}^{q} \ell_i^l$~~ for each subkernel, where r_l is the number of grid locations ~~(test inputs) x_i^* in the spatial~~ x^{ij} in the spatio-temporal grid such that for the l^{th} subkernel, ~~$k_l(x_i^*,x^*) < \sigma^2_{\text{min}}$. Here the parameters $k_l(x^{ij},x^*) \geq \sigma^2_{\text{min}}$.~~ For subkernels with $\gamma = 2$, $r_l \propto \prod_{i=1}^{q} \ell_i^l$, with ℓ_i^l ~~are the corresponding~~ denoting the length scale parameters over all the dimensions of the inputs x ~~this controls~~. In other words, r_l is proportional the size of the hypersphere inside which observations are considered for each x^*. The second bottleneck is calculating the Cholesky decompositions of the covariance matrices K with cost $\mathcal{O}((n_{\text{ker}}\kappa)^3)$. The cost of calculating the means and variances of the GP in a grid for a set of n_{times} points

on the time axis is therefore given by

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

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 ~~names of the subprograms here deviate from those in the code to improve readability.~~

~~The~~ function `AddToState()` reads observations (asynchronously) into a state object that tracks the proximity of each observation to each grid point. Only ~~part of data~~ a part of the observations is added, ~~and what part, is controlled on l.~~ 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

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

where ~~$d(x_i, x_{i_{\text{prev}}})$~~ $d(x^{\text{obs}}_i, x^{\text{obs}}_{i_{\text{prev}}})$ is the Euclidean distance of the point at x^{obs}_i that is being proposed for addition to the previous added point at $x^{\text{obs}}_{\text{prev}_i}$ and $\wedge$ is the standard notation for minimum. Hence with $\zeta = 0$ ~~, all observations will be~~ 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 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 lines 30-37, but with the ~~set of points x^*_i~~ inputs x^* in a random pattern instead of in a grid as is the case in l. 27-38.

The `AddSubdomainData()` method on l. 29 adds data as on lines 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. 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, that 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 2 can be increased for the wind-informed kernel and the value 8 is used in the demonstration ~~of the wind-informed kernel~~ in Sect. 4.8.

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

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)^{n_sd}_{i=1}) ← 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*, σ²_* ← ComputeMarginal(x*, A^obs_*);
33         if sampling = 2 then
34             ψ̂* ← Normal(μ*, σ²_*);
35             AddToState(state, x*,(ψ̂*, σ²_synthetic))
36         end
37     end
38 end
```

Figure 5. Overview of satGP execution. After initialization data is read for training m and k ~~, after which~~ and possible MRF computation is carried out. This is followed by sampling the prior if a synthetic study is performed, and learning the θ 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.

The Gaussian process algorithm is an interpolation algorithm when observation noise is zero, and interpolation algorithms may misbehave when used for extrapolation. In a spatio-temporal large grid, when sampling = 2, i.e. when draws of the Gaussian process are generated in a regular spatio-temporal 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. For this reason conditioning 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 are presented. In the first experiment , . The first experiment examines parameter identifiability with the multi-scale kernel is examined with using satGP-generated data. After that, the MRF of mean function β coefficients is trained with 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 data and those fields are then briefly analyzed.

Based on a 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 XCO2 spatio-temporal field are learnedfrom data. Knowing both the mean and the covariance functions , allows us to evaluate the Gaussian process is then solved globally in a grid and we present snapshots of the global mean and uncertainty fields are presented. The section is concluded by a concludes by comparing posterior marginal fields generated by using single-scale and multi-scale kernels and by demonstrating how the wind-informed kernel works. The covariance function parameters are learned from data.

4.1 Parameter identifiability with the multi-scale kernel

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

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

The covariance kernel parameters were still recoverable with a combination of three kernels ~~– Matern~~ , Matérn with $\nu = \frac{5}{2}$, exponential, and periodic~~, but for this, a larger κ was needed – the simulation shown used~~ . 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, or BOBYQA 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 four years according to the ~~values~~ true values reported in Table 3. A total of 10 million ~~Metropolis-Hastings iterations were carried out~~ 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

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	$\frac{\mathrm{est}-\mathrm{true}}{\mathrm{true}}$
τ^{mat}	0.05	1	0.5	0.652	0.304
$\ell^{\mathrm{mat}}_{lat}$	0.003	0.02	0.007	0.00989	0.413
$\ell^{\mathrm{mat}}_{lon}$	0.003	0.02	0.01	0.0135	0.350
ℓ^{mat}_{t}	1d	14d	7d	8.06d	0.15
τ^{per}	0.01	2	1	1.073	0.073
$\ell^{\mathrm{per}}_{lat}$	0.001	0.04	0.02	0.0207	0.035
$\ell^{\mathrm{per}}_{lon}$	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
$\ell^{\mathrm{exp}}_{lat}$	0.005	0.1	0.025	0.0352	0.408
$\ell^{\mathrm{exp}}_{lon}$	0.005	0.1	0.04	0.0405	0.0125
ℓ^{exp}_{t}	7d	30d	21d	24.83d	0.182

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 θ.

4.2 Posterior predictive distribution from synthetic WACCM4 ozone data

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

A synthetic study using WACCM4-generated ozone data was conducted to verify and to illustrate that the methods to learn the model parameters β, δ, and θ 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 with 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 $3.7\ \mathrm{k\,Pa}$ was selected.

Ozone data at approximately 400 locations were sampled daily over a two-year period in a random pattern from the domain 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 $12.5° \times 12.5°$ 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 δ, 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 $\hat{\theta}$. The values of the covariance parameters obtained were $\tau = 0.589$, $\ell_{lat} = 0.143$, $\ell_{\mathrm{lon}} = 0.225$, and $\ell_{\mathrm{t}} = 2\,\mathrm{d}\ 16\,\mathrm{h}\ 15\,\mathrm{min}$. 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 ψ^{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 one full year, and he total number of observations used was 39538. 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 ψ^{obs}.

The marginal posterior predictive distributions were computed globally in a uniform grid with the resolution of 2.5° in 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 Jan 6

2002 to Jan 5 2003, totaling around 4.384 million marginals in the predictive posterior. The one 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 Dec. 2 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 θ. The smallest reported uncertainties are close to zero, as they should, due to lack of observation error.

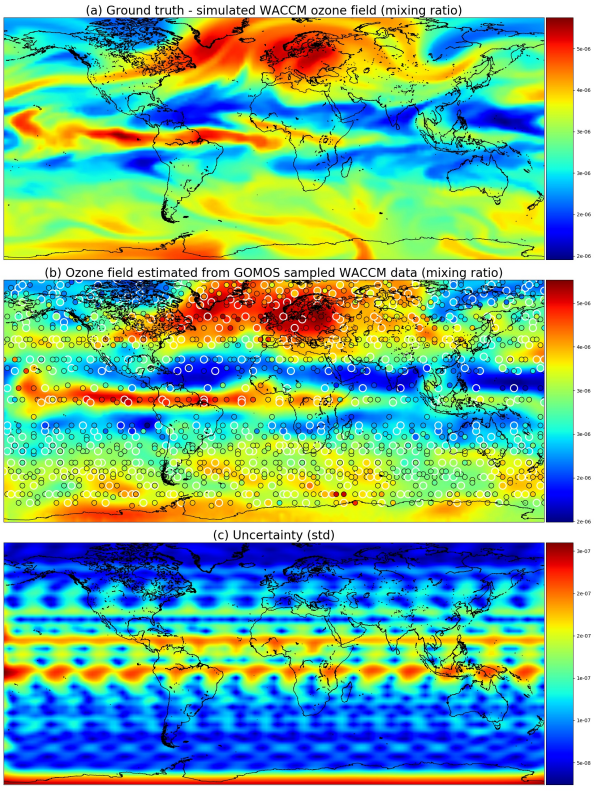

Figure 7. Ozone field mixing ratios at 3.7 k Pa for Dec. 2 2002. Panel (a) shows the simulated ground truth from WACCM4, (b) is the GP posterior mean, and (c) gives the posterior predictive uncertainties. A single Matérn kernel was used. In (b) the larger circles with the white edges are observations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

4.3 The OCO-2 v9 data

The simulations with ~~real~~ non-synthetic remote sensing data ~~utilize~~ use the v9 data from the OCO-2 satellite. ~~The~~ OCO-2 ~~satellite~~ 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 one day. The footprint area of each measurement is roughly ~~1.29 by 2.25 kilometers~~$1.29 \times 2.25\ \mathrm{km}^2$, but the data is very sparse in time and in space. ~~The satellite completes 14.57 revolutions around Earth overpasses in one day.~~ 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 ~~utilizes~~ uses the XCO2 data, its reported uncertainties, associated coordinate information, and zonal and meridional wind speeds that are contained in the data files. ~~Only observations flagged good are used, and~~ The time period considered is from Sept. 6 2014 to Nov. 10 2018 and we use only observations flagged as good, of which there are in total ~~116489342 such observations for the time period considered.~~ 116489342.

4.4 Solving the mean function for OCO-2 v9

~~Solving~~ Calibrating the mean function from OCO-2 v9 XCO2 data ~~,~~ as described in Sect.~~2.3.1, produces best~~ 2.3.1 produces the estimates for the ~~coefficients of Eq. (11)~~ β 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 ~~seasonality~~ variability of the XCO2 concentration~~, which~~. The timing of winter and summer in the Northern and Southern hemispheres explains the color shift along the equator. The lower left quadrant shows the amplitude of the twice faster oscillations, and like β_1, also β_2 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 are~~, and they are also areas where major emission sources are known to exist. ~~The observation of a local elevated concentration compared to the surrounding areas approaches the work of~~ Finding local elevated concentrations compared to surrounding areas echoes the observations made by Hakkarainen et al. (2016), where empirically defined time-integrated local XCO2 anomalies ~~are~~ were interpreted as possible emission sources.

The trend component β_4 varies only a little spatially, due to CO2 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 β factors. This partly explains the ~~slightly~~ different spatial pattern. The figure shows how the phases of the XCO2 annual cycles differ ~~in some regions, such as the Amazon or the Central African rain forests and the Sahel. The trend component β_4 was here set to be constant, as CO2 over time mixes in the atmosphere~~between regions, but the δ values need to be interpreted together with the β_1 and β_2 coefficients.

At high latitudes XCO2 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 x^{ij} to $\beta^{ij} \leftarrow \frac{\hat{\beta}^{ij} d}{20} + \frac{(20-d)\overline{\beta}}{20}$, where d is the distance to the edge of the domain in degrees, $\hat{\beta}^{ij}$ is the calibrated parameter vector at x^{ij}, and $\overline{\beta}$ is the average value of the parameters over the area where x^{ij} is located and where averaging is performed. The δ parameter was treated similarly. This adjustment was

done as a post-processing 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 XCO2.

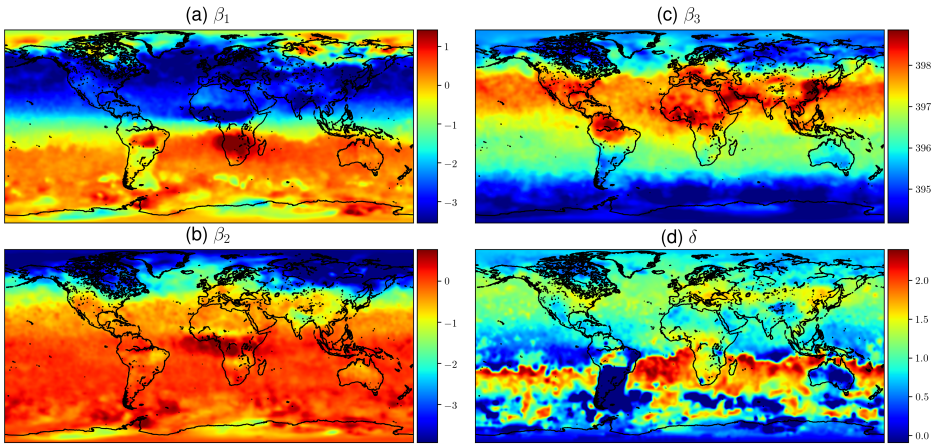

Figure 8. Mean values of mean function coefficients that were described as a Markov random field, calculated in a $2^\circ \times 2^\circ$ grid between 85° N and 85° S. The β_i coefficients multiply the f_i functions in Eq. (11). Panel (d) shows how the phase parameter δ can vary more in the southern hemisphere where β_1 and β_2 are small. The mean function and fitted daily means for several locations with the corresponding mean function parameters are shown in Fig. 1.

4.5 Covariance parameters of the OCO-2 v9 data

The OCO-2 data has several natural ~~length scales, both spatially and temporally~~spatial and temporal length scales. The distance between adjacent observations is only one to two kilometers 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 three hundred kilometers 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 ~~in the particular form of~~ by finding the mean function ~~used, Eq.~~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~~, and the data is~~. The OCO-2 data is therefore modeled with a kernel consisting of a larger-scale exponential and ~~smaller scale Matern component~~a smaller scale Matérn subkernel.

The covariance parameters for the two-component kernel ~~, which are~~ are given in Table 4. The values used were the median values from sampling the posterior with MCMC~~, are given in Table 4. With~~. When learning the parameters from a data set with several natural length scales, the posterior may appear multi-modal, 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.

Table 4. Covariance function parameter values learned from OCO-2 data. First column shows the ~~Matern~~ Matérn kernel parameters, and the second column the exponential kernel parameters. The length ~~scale~~ scales along the parallels, $\ell^{(\cdot)}_{\text{lon}}$ is much larger than that along the meridians, $\ell^{(\cdot)}_{\text{lat}}$.

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

4.6 Posterior predictive distributions of XCO2 from the OCO-2 v9 data

The marginal posterior predictive distribution at test points x^*, given by Eq. (9) and (10), were calculated globally in a half-degree grid between 80°S and 80°N at ~~a daily~~ daily time resolution. The first day of simulation was September 6 2014, and the last day was November 10 2018, spanning in total 1526 days. For each day, 230400 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 time was ~~25~~ 26 days on a moderately fast Intel i7-8700K CPU utilizing the available 12 CPU threads and 32 GiB memory.

~~Global~~ Figures 9 and 10 present global fields of the mean values and marginal uncertainties~~are presented in Fig. 9 and 10~~, with a subset (to avoid excessive over-drawing) of observations shown as a scatter plot ~~. For this simulation, a maximum distance of 1100 km (10° on the equator) was specified for speeding up searching for closest observations in the direction along parallels. This constraint can be seen as discontinuities in uncertainty when no observations are nearby, especially close to the poles~~in the (a) panels. The (b) ~~parts of the figures~~ panels show how uncertainty is reduced with the overpass of OCO-2. This uncertainty reduction diminishes fast due to the ~~Matern~~ Matérn component of the multi-scale kernel having a very short length scale parameter in the time dimension. In the upper figures ~~,~~ the background color (~~posterior mean~~ mean of the Gaussian process posterior) usually matches the observations~~. Due~~ , but due to observational noise, the ~~GP~~ posterior mean is not ~~strictly interpolation, however~~everywhere an interpolated field.

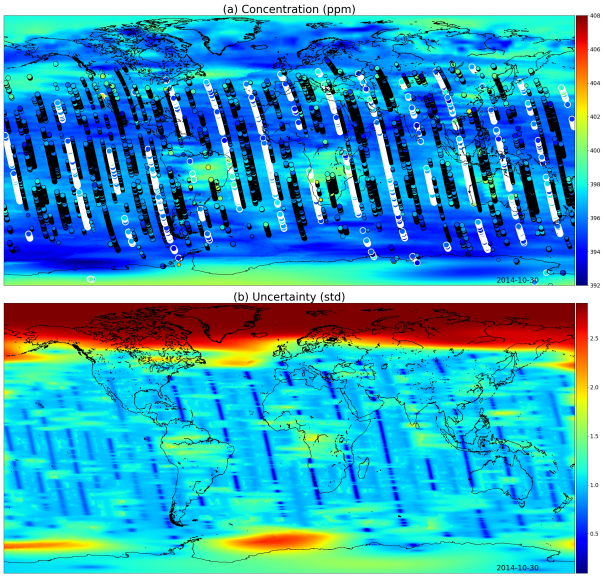

Figure 9. ~~Global XCO2-distribution~~ XCO2 posterior mean values (a) and their uncertainties (b) on ~~last day~~ the 30$^{\text{th}}$ of October 2014. The most informative observations are shown with the concentrations, with the large white circles being from ~~October 31$^{\text{st}}$ 2014~~the 30$^{\text{th}}$, medium circles from one day before or after, and small circles from two days before or after. The OCO-2 utilizes sunlight for retrieval, ~~and that~~ which is why there are very few observations above 60°N. These fields include latitudes up to 85°S and 85°N.

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

~~How~~ Data from the OCO-2 can be used to demonstrate how the multi-scale kernel formulation affects the predictive posterior distributions~~can be demonstrated with OCO-2 data. In Fig. 11~~. Figure 11 shows posterior marginals from September 15 ~~2014 are shown.~~ 2014. The first row (a-b) contains results from the multi-scale kernel described in Sect. 4.5, and the second row (c-d) shows fields from only the exponential part of the multi-scale kernel. The parameters of the multi-scale kernel are shown in Table 4. The bottom row (e-f) contains the difference fields between the first and the second rows. The single-kernel uncertainty is very low in Fig. 11 (d) since lots of observations fall into regions of high covariance with almost any test input, with the exception of the ~~Northern~~ 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 XCO2 field best.

~~Figure~~ Panel (a) shows that as intended, the multi-scale approach leads to local enhancements of the XCO2 mean field. Far from the measurements, the smaller ~~Matern~~ Matérn kernel no longer reduces the predicted marginal uncertainties, and this leads to an increase in uncertainty in these areas. Figure (e) shows additional enhancements of the XCO2 mean fields, which are in this case due to the different maximum covariances between the multi-scale and single-scale kernels.

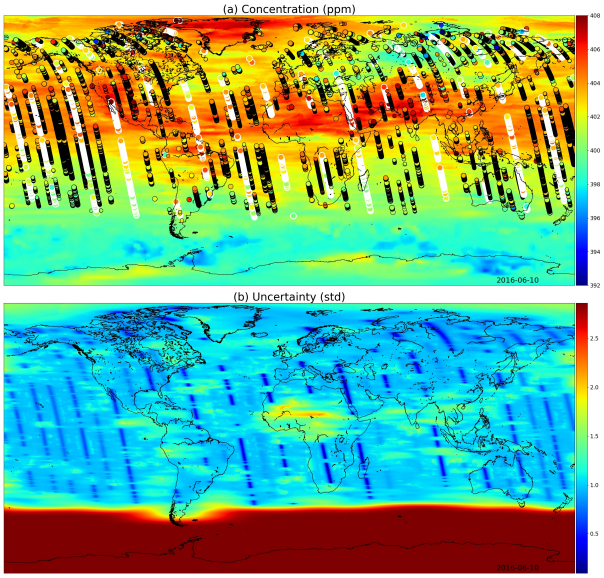

Figure 10. ~~Global XCO2-distribution~~ XCO2 posterior mean values (a) and their uncertainties (b) on 10^{th} of June ~~1^{st}~~ 2016. While photosynthesis in the Northern Hemisphere is already reducing the carbon dioxide concentrations globally, the observations condition the Gaussian process to higher 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.

The total kernel size was kept at 1024 ($\kappa = 512$ for (a-b) and $\kappa = 1024$ for (c-d)) in both experiments~~. Additionally~~, and thinning and grid resolution parameter values were $\zeta_{\text{sample}} = 5$, and $\omega = 0.5°$~~in this case~~. The very same observations were used ~~in both cases~~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 x^* rotate and scale the ~~coordinate axes~~ 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 of 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 2.652s (walltime) on the i7-8750H laptop CPU.

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

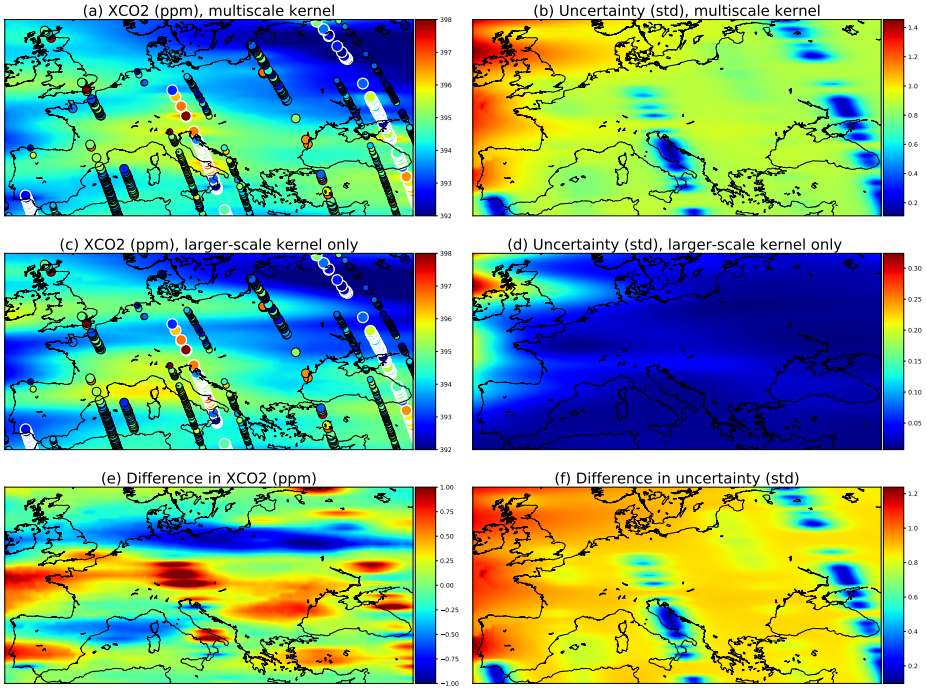

Figure 11. Comparison of a multi-scale kernel with the two components described in Sect. 4.5 and a single component kernel defined by the parameters of the exponential kernel. These parameters were given in Table 4. The observations used are the same and are shown in panels (a) and (c) as circles. The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

Optimally the wind-informed kernel should utilize winds that are not recomputed from the observations as was done here for convenience, but directly from a weather or climate model or from a wind data product. The satGP program 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 ~~have~~ introduced the first version of a ~~new fast~~ fast general purpose Gaussian process software, satGP ~~v.0~~v0.1~~. It aims at being a general purpose Gaussian process toolbox, especially meant~~ .2, which is in particular intended to be used with remote sensing data. ~~The software solves the problems of~~ We showed how the program solves spatial statistics problems of enormous sizes by using a spatially varying mean function, ~~learning its parameters via computation of~~ learned by computing marginals of an MRF, and ~~also allows learning the parameters of the~~ by using a multi-scale covariance function~~using either~~ , parameters of which are found either by using optimization algorithms or with adaptive Markov chain Monte Carlo. ~~On top of these, satGP allows to conduct~~ We also presented how satGP allows conducting synthetic parameter identification

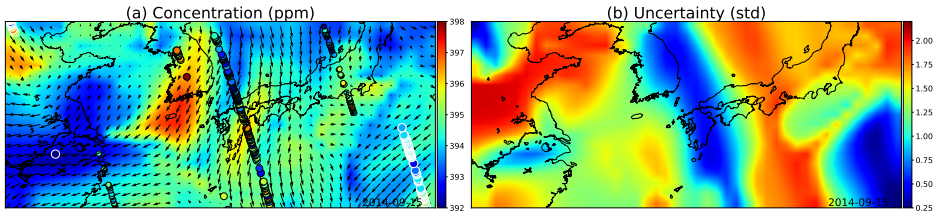

Figure 12. (a) GP posterior mean of XCO2 and (b) its uncertainties with the wind-informed kernel. The area shown contains the Korean peninsula in the center, China on the left, and Japan on the center-right. The large circles with the white edges are present-day observations, medium circles are observations from adjacent days, and the smallest ones are observations from two days 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.

studies ~~via sampling~~ by sampling from Gaussian process prior and posterior distributions, and this ~~can~~ could be done with any kernel prescribed, including a non-stationary wind-informed kernel. ~~We are not aware of open source remote sensing-oriented software that would provide this combination of features. The satGP program was demonstrated with the enormous~~ The features of satGP were demonstrated first with a small scale synthetic ozone study, and then using the enormous XCO2 data set produced by the NASA Orbiting Carbon Observatory 2.

~~There are various~~ Various aspects of satGP ~~that could~~ can be improved in future versions~~. These include addition of routines for doing model selection to select the components of the multi-scale kernel,~~ , some of which include improving the observation selection/thinning scheme for statistical optimality, ~~and finding joint posterior predictive distributions. For the last one, a multi-grid version can be developed, and this could be potentially useful for flux inversion studies.~~

~~The satGP software utilizes various approximations for computational tractability, and the connection between parameters such as length scales ℓ, thinning parameter ζ, maximum kernel size κ, and prediction accuracy could be studied further, as well as changing the grid resolution according to density of observations.~~

~~The methodology and code presented can be also used with other data sources.For instance, combining data from the various satellites that measure CO2,~~ 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 ~~the~~ OCO-3~~, would be particularly interesting. That more and more instruments are about to provide data from the orbit in the near future will lead to a need to understand the properties of even larger data sets~~. 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 analysis.

Code and data availability. The satGP code is currently available from the corresponding author upon a reasonable request, and we will release the software under the open source MIT license as a supplement to the final version of this manuscript. 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.

Appendix A: Input parameters and variables in satGP

The satGP software by design allows for ~~lots~~ a lot of flexibility for defining how to model the quantity of interest as a Gaussian random field. ~~In this section the possibilities are discussed along with some~~ 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`. ~~Practical aspects of defining mean functions and covariance kernels are also included.~~ Some of the details in this section may change for future versions of ~~satGP~~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 learn the covariance function, and only after that calculate the means and variances of the Gaussian process with ~~-~~sampling $= 1$. The setting ~~-~~sampling $= 2$ can be used e.g. for illustration purposes, ~~for~~ 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 ~~is wished to be used~~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 ~~hardcoded~~ hard coded in the code ~~base to~~ 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. These 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 ~~need also~~ 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 ~~at~~ 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 ~~also~~ more computational cost, due to the κ parameter, which is the last one in the table and is discussed later.

The parameters ~~and~~ cfc and mf are not strictly input variables, but 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. ~~Matern~~Matérn, exponential, and periodic), and then defining the parameters for those kernels. The best parameters are those that are learned with ~~learn~~ learn_k $= 2$ when non-synthetic data is used.

~~The learning~~ Learning the covariance parameters θ is best performed with MCMC, and the posterior mean and median have proven to be ~~a~~ useful values. For unimodal posterior distributions these values are ~~very close~~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~~, and for computing the log-likelihood in Eq. (24), the~~ . 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. 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 ~~Matern~~ Matérn kernel needs to be changed, that can be done by changing the variable `n` in functions `covfun_matern52()` and `initialize_covfunconf` in that same file.

For constructing the mean function, the configuration file contains the parameter `mftype`. The possible values are: 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 (β and δ in Eq. (2)) if `mftype` $\in \{1,3\}$. If the mean function parameters are learned, the parameter `nnonbetas`, the number of mean function non-linear δ 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 coefficients, the values of those coefficients are given in the configuration file~~. Additionally,~~ , where the parameter limits for learning the space-dependent mean function parameters are ~~set in the configuration file~~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 β parameters is done in section 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} ~~parameter has the same effect for when~~ 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. ζ. 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 ~~also~~ increased if faster execution is required, ~~e.g.~~ for example for debugging purposes.

The σ^2_{min} parameter controls which observations are not considered at all when computing at a location x^*, as described by Eq. (21). The higher this is, the more data is 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 ~~actually~~ meant to be changed ~~, and it is for that reason~~ often, due to which 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 ~~settings~~ setups with only a single subkernel. If $\sigma^2_{\text{synthetic}}$ is high, parameter identifiability suffers. ~~Varying this parameter could be used for understanding how complex a multi-scale kernel can be useful with particular data sets. The values also depend~~ 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 is included for constructing the covariance matrix 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 speed-up due to solving the GP formulas faster decreases, since at that point computing Cholesky decompositions no longer takes up ~~majority of~~ 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 non-zero, since with random selection adjacent grid points often do not utilize the best available observations closest by, leading to noisiness or graininess in the ~~computed~~ 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 GCC 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.

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

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

Acknowledgements. This work was supported by the Centre of Excellence of Inverse Modelling and Imaging (CoE), Academy of Finland, decision number 312122. 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). Copyright 2020. All rights reserved. US Government Support Acknowledged.

References

Ambikasaran, S., Foreman-Mackey, D., Greengard, L., Hogg, D. W., and O'Neil, M.: Fast Direct Methods for Gaussian Processes, IEEE Transactions on Pattern Analysis and Machine Intelligence, 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, Advances in Space Research, 33, 1029 – 1035, https://doi.org/https://doi.org/10.1016/j.asr.2003.09.037, http://www.sciencedirect.com/science/article/pii/S0273117703008007, climate Change Processes in the Stratosphere, Earth-Atmosphere-Ocean Systems, and Oceanographic Processes from Satellite Data, 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, Atmospheric Chemistry and Physics, 10, 12 091–12 148, https://doi.org/10.5194/acp-10-12091-2010, https://www.atmos-chem-phys.net/10/12091/2010/, 2010.

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

Cressie, N.: Mission CO2ntrol: A Statistical Scientist's Role in Remote Sensing of Atmospheric Carbon Dioxide, Journal of the American Statistical Association, 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., Elder-ing, 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 XCO2 data characterization, Atmospheric Measurement Techniques, 5, 687–707, https://doi.org/10.5194/amt-5-687-2012, https://www.atmos-meas-tech.net/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, Journal of the American Statistical Association, 111, 800–812, https://doi.org/10.1080/01621459.2015.1044091, 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, Atmospheric Measurement Techniques, 12, 2341–2370, https://doi.org/10.5194/amt-12-2341-2019, https://www.atmos-meas-tech.net/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, http://www.jstor.org/stable/3318737, 2001.

Hakkarainen, J., Ialongo, I., and Tamminen, J.: Direct space-based observations of anthropogenic CO2 emission areas from OCO-2, Geophysical Research Letters, 43, 11,400–11,406, https://doi.org/10.1002/2016GL070885, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2016GL070885, 2016.

Hammerling, D. M., Michalak, A. M., O'Dell, C., and Kawa, S. R.: Global CO2 distributions over land from the Greenhouse Gases Observing Satellite (GOSAT), Geophysical Research Letters, 39, https://doi.org/10.1029/2012GL051203, https://agupubs.onlinelibrary.wiley.com/doi/abs/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, Journal of Agricultural, Biological and Environmental Statistics, 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, Bulletin of the American Meteorological Society, 94, 1339–1360, https://doi.org/10.1175/BAMS-D-12-00121.1, https://doi.org/10.1175/BAMS-D-12-00121.1, 2013.

IPCC: Summary for Policymakers, book section SPM, pp. 1–30, Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, https://doi.org/10.1017/CBO9781107415324.004, www.climatechange2013.org, 2013.

Johnson, S. G.: The NLopt nonlinear-optimization package, http://github.com/stevengj/nlopt, 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, Advances in Space Research, 33, 1020 – 1028, https://doi.org/https://doi.org/10.1016/S0273-1177(03)00590-8, http://www.sciencedirect.com/science/article/pii/S0273117703005908, climate Change Processes in the Stratosphere, Earth-Atmosphere-Ocean Systems, and Oceanographic Processes from Satellite Data, 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, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73, 423–498, https://doi.org/10.1111/j.1467-9868.2011.00777.x, https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.00777.x, 2011.

Ma, P. and Kang, E. L.: Fused Gaussian Process for Very Large Spatial Data, ArXiv e-prints, 2017.

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), Journal of Climate, 26, 7372–7391, https://doi.org/10.1175/JCLI-D-12-00558.1, 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 CO2 Emissions From Individual Power Plants From Space, Geophysical Research Letters, 44, 10,045–10,053, https://doi.org/10.1002/2017GL074702, https://agupubs.onlinelibrary.wiley.com/doi/abs/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, Atmospheric Measurement Techniques, 5, 193–193, https://doi.org/10.5194/amt-5-193-2012, https://www.atmos-meas-tech.net/5/193/2012/, 2012.

Rasmussen, C. and Williams, C.: Gaussian Processes for Machine Learning, MIT Press, http://www.gaussianprocess.org/gpml/chapters/, 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, first 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, Geoscientific Model Development, 10, 709–720, https://doi.org/10.5194/gmd-10-709-2017, https://www.geosci-model-dev.net/10/709/2017/, 2017.

Vecchia, A. V.: Estimation and Model Identification for Continuous Spatial Processes, Journal of the Royal Statistical Society. Series B (Methodological), 50, 297–312, http://www.jstor.org/stable/2345768, 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, http://www.eecs.berkeley.edu/~wainwrig/Papers/WaiJor08_FTML.pdf, 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 Journal of Space Science, 38, 627, https://doi.org/10.11728/cjss2018.05.627, http://www.cjss.ac.cn/EN/abstract/article_2600.shtml, 2018.

Yokota, T., Yoshida, Y., Eguchi, N., Ota, Y., Tanaka, T., Watanabe, H., and Maksyutov, S.: Global Concentrations of CO2 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, Chemometrics and Intelligent Laboratory Systems, 149, 227 – 241, https://doi.org/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, https://doi.org/10.3390/rs10010155, http://www.mdpi.com/2072-4292/10/1/155, 2018.

Zeng, Z., Lei, L., Guo, L., Zhang, L., and Zhang, B.: Incorporating temporal variability to improve geostatistical analysis of satellite-observed CO2 in China, Chinese Science Bulletin, 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 CO2 total columns using spatio-temporal geostatistics, International Journal of Digital Earth, 10, 426–456, https://doi.org/10.1080/17538947.2016.1156777, 2017.