Reviewer comments are in *black italics.* Author responses are in orange, with new additions to the manuscript in **bold orange.**

Review: Using a Gaussian Process Emulator to approximate the climate response patterns to greenhouse gas and aerosol forcings (egusphere-2025-6046)

1 Overview

This manuscript presents a statistical model in the form of a Gaussian Process (GP) emulator to approximate the surface temperature response as a function of a collection of climate forcings including global greenhouse gas concentrations and regional aerosol emissions. The GP model is built on strong statistical theory and methodology. It yields substantial computational efficiency gains compared with using a Global Climate Model (GCM) directly. The model is tested and applied to the HadGEM3 GCM (for which a good description of the setup is provided) where it is employed to perform a sensitivity analysis, as well as demonstrating how it may be used for policy-assessments under rapid climate change projections. The results are consistent with those obtained in other studies such as the sixth Coupled Model Intercomparison Project (CMIP6), and using reduced complexity models. Furthermore, given the model description and the provided Python code, it would be possible to apply the GP emulator to other GCMs.

This review focuses on the modelling aspect of the manuscript. The research is within the scope of Geoscientific Model Development (GMD) with the application illustrating its use to answer scientific questions of interest to EGU. In addition, the presented methodology constitutes a novel contribution to statistical modelling of climate models by aiming to model spatial patterns rather than the global mean response with respect to time, as well as the capability to incorporate a larger number of climate forcings as input parameters.

The manuscript is logically structured in its presentation of the GCM, input parameters, GP emulator, model training data, and its application and evaluation. Justification is provided throughout for the modelling choices and assumptions explaining potential limitations and how these are addressed. The title clearly describes the content of the manuscript. It is also well written, uses good quality and necessary figures to illustrate the results, contains appropriate references to existing literature, and the abstract provides a good summary.

In summary, this manuscript presents a new form of statistical model as an approximation to output of a GCM which provides an efficient means of performing important uncertainty quantification tasks such as sensitivity analyses and policy assessments. I recommend publication subject to a few minor revisions.

We would like to thank the referee for their time spent reading our manuscript and providing helpful comments which will improve the quality of the paper. We are glad that the reviewer has a positive view of our work and recommends publication of the manuscript following minor revisions. We address the referee's comments as we outline below.

2 Main Points and Suggestions

Below I outline several suggestions to enhance the presentation of the model and clarity of the manuscript.

2.1 Univariate versus Multivariate Emulation

The aim of the GP emulator presented in this manuscript is to predict the entire spatial pattern of the global climate change response. In section 2.2 it is outlined that the Earth is split into N grid cells with independent GP emulators constructed for each grid cell. This is a common approach which has been successfully employed in other climate modelling studies, for example, Johnson et al. (2020) and Lee et al. (2012). However, there is a risk that the emulator mean predictions and their uncertainty for neighbouring grid cells are spatially inconsistent. It is observed in figure 1 and in Supplementary figure S4 that for a variety of input settings that this is not the case in this application, although it is plausible that using a different GCM, training data, or input parameters, that such issues may occur.

Have the authors considered multivariate GP emulators? There exists wide ranging methodology with many having been employed in applications to climate modelling. Some examples include: separable GPs in Conti et al. (2010) and Higdon et al. (2008); outer-product GP emulator in Rougier (2008) and Rougier et al. (2009); dimension reduction of the spatial field, such as in Salter et al. (2022, 2019) and Wilkinson (2010); or parallel partial GP emulators in Gu et al. (2016, 2019). These incorporate extra spatial structure information, thus ensuring spatial consistency, and with the potential for smaller uncertainties through the sharing of information and patterns between grid cells. Moreover, in the spatial statistics literature the Kriging methodology is well established and amounts to GP emulation of spatial fields (see for example Cressie (1993)) with specific covariance functions, such as the C4-Wendland class of correlation functions, designed to incorporate spatial inputs on spheres (Gneiting (2013)). It is acknowledged that these methods are generally result in more complex emulators to formulate and can be more computationally expensive to train, although this is often negligible compared with the cost of running the GCM. There is a trade-off to be achieved with the emulator accuracy. For this manuscript, what is the justification for N independent emulators versus a multivariate approach?

We did consider multivariate GP emulators, however, we found them to be computationally prohibitive to train due to the large number of grid cells, and we found that the independent emulators performed well for our task. We did also explore dimension reduction such as PCA to induce appropriate correlations on the spatial field and found this to give similar results to those presented in our manuscript.

We add a comment on this in the methods:

Line 230:

> "Here, we use independent emulators for each grid-cell, **however, we also found similar results with GP emulators applied to the top principal components, a dimension reduction technique that also enforces spatial correlations between grid-cells (e.g., Salter et al., 2019; Salter & Williamson, 2022; Wilkinson, 2010)**."

We appreciate the suggestion that multivariate GPs may be able to share spatial information, thereby reducing uncertainties and we also comment on this as an extension in the conclusions:

Line 729-731:

> "**Another approach to reducing uncertainties could be to develop multivariate GP emulators which account for the spatial structure and may be able to share information between neighbouring grid cells (e.g., Conti & O'Hagan, 2010; Gu et al., 2019; Gu & Berger, 2016; Higdon et al., 2008; Rougier, 2009; Rougier et al., 2009).**"

In the application to HadGEM3, how many grid cells are there and hence what is N? This will aid the reader in understanding whether this methodology would be applicable to their model where for larger values of N when using a higher resolution model this adds to the computational expense of fitting, testing, and evaluating the emulator.

Updated to include number of grid cells:
Line 292: "Here, we use independent emulators **for each of the 27,648 grid cells**"

2.2 GP Emulator Structure
Section 2.5 describes the GP emulator structure. In line 226 it is stated that "the prior mean function is often assumed to be zero", for which it is implicit that this is the assumption made throughout this manuscript. An alternative approach is to build more global structure into the mean function, such as using low order polynomials of the input parameters, thus leaving the covariance kernel to handle the smaller scale

local variability, such as in Craig et al. (1996), Edwards et al. (2019), and Vernon et al. (2010). If the underlying mean behaviour is constant with respect to the inputs, then emulators formulated in such a way are still capable of identifying them. Have the authors considered a structured mean function? Would this enable building more accurate emulators? The choice of variables to incorporate in a structured mean function may also be driven by the results shown in the sensitivity analysis where the global CO2 concentration forcing parameter has a large sensitivity index and thus may be informative as a term in the mean function.

In Line 226, we explicitly say 'as done here' to clarify that this assumption is made: "The prior mean function is often assumed to be zero so that all choices are determined by the covariance function, as done here."

We have not extensively explored building global structure into the mean function, although we have used a non-stationary linear covariance function with ARD to determine the slope independently for each input. This should be equivalent to a linear mean function as the reviewer describes and we comment on this.

Line 419:

> "**This captures the first-order linear relationships between emission perturbations and temperature response, while the remaining terms aim to capture the local variability (Craig et al., 1996; Edwards et al., 2019; Vernon et al., 2010).**"

2.3 Covariance Kernel

The construction of the covariance kernel consists of the sum of a linear kernel, a squared exponential kernel, and a white noise process (also known to as a nugget term). However, it is unclear the exact structure this refers to, and the associated hyperparameters. Examples of ambiguities include:

1. Whether it is a homogenous or an inhomogeneous linear kernel,

2. The exponent form in the squared exponential kernel such as using either a single common correlation length (also known as length scale) hyperparameter, separate and potentially distinct correlation lengths, or encompassing normalisation of parameter vector differences via an input variance matrix (the fully anisotropic form of the squared exponential covariance function),

3. Whether a separate variance hyperparameter is used for each additive component, or if a shared variance hyperparameter is employed, potentially with an additional weighting hyperparameter between the two components.

This could be resolved by including the mathematical formula for K(x, x′), highlighting each of the hyperparameters which are estimated via Maximum Likelihood Estimation.

In this formulation, does σGCM correspond to the white noise variance hyperparameter? How does σGP relate to the variance hyperparameter of the non-white noise part of the kernel? Also, is there some intuition for why this additive combination works for the surface temperature response being modelled, for example, due to the different properties of the kernels? Please clarify these details. Furthermore, it is initially unclear that σGP and σtotal are mean uncertainties over the 18 test scenarios. It would be helpful to clarify this and remind the reader that the GP emulator's uncertainty depends on the new input settings at which it is evaluated, and that σGP and σtotal are summary statistics for these. Investigations shown in Supplementary figure S3 illustrate similar results irrespective of which of a variety of covariance kernels are used in the emulator formulation. Why did the authors opt for the more complex kernel described, rather than for one with fewer hyperparameters, such as MatÅLern-5/2, which is commonly employed across climate modelling? Also, what is the intuition for why this form of covariance kernel works? Can certain modes of variability be associated to each component of the covariance function?

We thank the reviewer for pointing this out and have clarified the full covariance structure in Section 3.3, see below. This also includes some descriptions of the meaning of the hyperparameters, and that the white noise kernel used is either fixed based on the GCM values or is a learnable hyperparameter.

Line 270:

> The kernel defines the similarity of two inputs $\boldsymbol{x}$ and $\boldsymbol{x}'$ and how this propagates through to the similarity of the outputs $\boldsymbol{f}_{\boldsymbol{GP}}(\boldsymbol{x})$ **and** $\boldsymbol{f}_{\boldsymbol{GP}}(\boldsymbol{x}')$**. We use**
>
> $$K(\boldsymbol{x},\boldsymbol{x}') = \sum_{d=1}^{9} \sigma_{Ld}^2 \, x_d \, x_d' + \sigma_{SE}^2 \exp(-\frac{1}{2}\sum_{d=1}^{9}\frac{(x_d - x_d')^2}{l_d^2}) + \sigma_W^2$$
>
> **This describes an additive kernel consisting of three components. The first term represents a non-stationary linear kernel which learns independent hyperparameters, $\boldsymbol{\sigma_{Ld}^2}$, for each input parameter, $\boldsymbol{d}$ (known as Automatic Relevance Determination, ARD), representing the rate of change of the output for each input parameter. This captures the first-order linear relationships between emission perturbations and the temperature response, while the remaining terms aim to capture the local variability (Craig et al., 1996; Edwards et al., 2019; Vernon et al., 2010). The second term is a squared exponential kernel which also uses ARD to learn independent lengthscales $\boldsymbol{l_d}$ for each input parameter, $\boldsymbol{d}$. These lengthscales describe how rapidly the output changes with respect to varying input parameters. The hyperparameter $\boldsymbol{\sigma_{SE}^2}$ is a scale**

> **that describes the total amount of variation in the output (Rasmussen & Williams, 2006). The last term is a white noise kernel which represents the internal variability of the GCM, assumed to be constant, σ_W^2, for all predictions. We explored both fixing σ_W^2 at the internal variability derived from the GCM (described further below and used throughout the paper unless stated otherwise), and allowing σ_W^2 to be learned as an additional hyperparameter (described in Section 3.3). We chose this kernel function because of its interpretability, however,** we found the emulator to be robust to a range of kernel choices (Supplementary Fig. S3).

Following the reviewer's comments, we also comment on which terms are dependent on *x* on Line 315 which discusses the internal variability component of the noise:

Line 315:

> "We expect this to be present for any prediction and therefore, we include this as a fixed noise term in our emulator, **by setting $\sigma_W^2 = \sigma_{GCM}^2$ for the white noise covariance function of the GP emulator.** When the emulator predicts the temperature response at new unseen input values, there is also predictive uncertainty provided by the GP emulator, which we label σ_{GP}. **Note that this comes from the other covariance functions and is dependent on the inputs, x.** The total uncertainty arising from both the GCM internal variability and the GP emulator prediction is denoted σ_{total} and these are related via ${\sigma_{total}}^2 = {\sigma_{GCM}}^2 + {\sigma_{GP}}^2$, discussed later in Section 3.3."

In section 3.3 and in Supplementary figure 5 it is discussed how similar uncertainty estimates for σtotal and σGPlearn are obtained where the latter does not use the fixed estimate of σGCM. Please add information on whether the two emulator structures are otherwise identical. In particular, does the second emulator possess a white noise process for which the variance hyperparameter is estimated simultaneously with all other emulator hyperparameters? This is important for a fair comparison and interpretation of the results. In addition, reporting and contrasting the hyperparameter estimates may divulge why σtotal and σGPlearn are similar, as well as provide an explanation for the main sources of the GP emulators predictive uncertainty.

We do use the same emulator structure with a white noise process and our edits to the kernel structure described above should address this comment and clarify that the emulator structure remains the same. In section 3.3, we add:

Line 405: "For comparison, we built an alternative version of the GP emulator**, where rather than fixing the internal variability based on the GCM internal variability, $\sigma_w^2 = \sigma_{GCM}^2$, we treat the white noise covariance σ_w^2, as an additional hyperparameter to be optimized."**

A related minor point is in the learning of σGCM. In lines 242 – 250 this is estimated based on 14 control runs, 8 of which are obtained by splitting a 40-year control run into 8 non-overlapping segments. It is rightly acknowledged that the sample size is small and that the autocorrelation is likely to have an effect. A potential consideration would be to adjust the estimate of σGCM to account for the autocorrelation in the variance estimate calculation.

We appreciate this point, but we do not pursue this here for simplicity. We have checked that autocorrelation between the 8 simulations is relatively weak, falling between -0.3 and 0.3 for most grid cells for the lagged 1 to 4 autocorrelations functions We also find similar patterns if we compute σ_{GCM} over the 6 independent control runs only.

2.4 Training Design

In section 2.3, lines 184 – 193, the design construction mechanism is described. An interesting feature is the split where 75% (60 simulations) are sampled within the parameter ranges given in table 1, whilst the remaining 25% (20 simulations) are sampled from extended parameter ranges with the 7 regional aerosol forcing parameters upper range doubled. The justification seems reasonable based on improving signal-to-noise ratio for enhanced emulator predictive fit. Linking to section 2.4, where it is stated that a maximin Latin hypercube sampler is implemented, it is unclear how this weighted sampling from a nested parameter space is achieved. Please provide more details as to the sampling algorithm. For example:

- *Are two separate maximin Latin hypercube designs over each space constructed (potentially with rejection of points from the larger space)?*

We thank the reviewer for highlighting that this was unclear in the manuscript. We do not use two separate Latin hypercube designs. The "75%-25%" split was not clearly defined, as we do not sample 60 simulations from within the parameter space and 20 in the extended range. Rather, we mean that the distribution for the parameters ensures that 75% of the samples are within the defined range, and the remaining 25% are spread further across the parameter space. This means that the points are more spread out for the larger forcings. We clarify in the text:

Line 255-262:

> "**To generate the samples with the distribution described above, we first sample 80 points using a maximin LHS for values between 0 and 1 for all inputs. For the GHGs, we use a linear transform to scale the input range so that all values fall within the defined bounds. For the short-lived pollutants, we use a linear transform to scale the first 75% of samples to lie within the ranges defined in Table 1. We then scale the remaining 25% of samples to fall between 1 and 2X the maximum values in Table 1. This provides us with samples that are further separated for the larger perturbations values, but more closely sampled in the region we are most interested in.**"

• Is there a joint optimisation (maximisation of inter-design point distance) across the two components of the design to ensure that points outside of the standard parameter range are not too close to a design point within the range?

No, we use a standard LHS design for simplicity. We clarify:

Line 261: "We also perform an additional 18 simulations for testing, which are sampled separately from the ranges in Table 1 using **a standard LHS design**"

Further questions raised by the design approach are:
1. What is the reason for using a 75-25 split? The extended parameter space possesses a volume 27 = 128 times that of the standard parameter space, yet only a quarter of the design points are sampled from this reason.

We hope that our new description of the sampling approach clarifies this. We do not sample 20 training points from the extended region only (which would mean that all 20 samples would have unrealistically large parameter perturbations), but instead we use a scaling to ensure that 25% of samples come from the extended region *for each input parameter*. This means it is fairly likely that each training point has one large parameter perturbation.

2. Why is the range doubled? This is particularly important if the aim is to have an accurate emulator over the standard parameter space, and given how much larger this extended parameter space is. Moreover, if it is reasonable to believe that aerosol forcings may be large, why is the input parameter range not extended to account for this?

The range is doubled to increase the signal to noise ratio in some of the perturbations and to help the emulator learn the signals. Even with these larger perturbations, we see relatively weak signals compared to the GHG forcing perturbations.

Line 225-230 (not changed):

> "The fact that we apply regional perturbations to the short-lived pollutants inevitably leads to small changes in abundances and weak forcings. This can give weaker grid-scale climate response signals that are difficult to separate from the internal noise (Kasoar et al., 2018; Shawki et al., 2018). For this reason, we extend the input space beyond the ranges described in Table 1 for these forcers, in order to create an emulator that gives less uncertain results. We expect that including simulations with stronger perturbations, which should have larger signal-to-noise ratios, could be beneficial for constraining the emulator prediction to these perturbations. We therefore extend the input space for the short-lived pollutants only so that 75% of simulations fall within the input range indicated in **Error! Reference source not found.** and the remaining 25% lie outside this but within a range that is double this (see Supplementary Fig. S2)."

3. How is fitting emulators using the wider parameter range reflected in estimates of correlation length parameters? If design points in the extended parameter space are a long distance with respect to the correlation lengths from the nearest point at which the emulator is evaluated then they will provide little extra information.

We have looked at the emulator parameters in more detail. The RBF correlation lengths vary significantly between grid cells. There are some regions where the correlation lengths are short suggesting that the additional points do add extra information. We also expect that the wider parameter range help other kernel parameters, including the linear parameters which are estimated to have larger values over the emission region for the aerosol perturbations.

4. What is the additional cost of running the 20 simulations outside the standard parameter ranges, and is this justified by the improvements in emulator predictive uncertainty? In an ideal scenario, although not practical here due to the cost of obtaining further simulations from HadGEM3, a fair comparison would be the emulators obtained using the presented design with an emulator fitted to an 80-point maximin Latin hypercube design over the standard parameter space only.

The reviewer makes an interesting point but we have not explored the additional costs involved or the gain in emulator accuracy. This would be an interesting topic for future work, which we now comment on this in the conclusions:

Line 733-759:

> **The overarching goal of this study was to identify a framework for policy-relevant emulators. We made several choices when designing the training and test dataset using GCM simulations, from which future studies can learn. For instance, we chose a maximin Latin hypercube design to sample uniformly across the distribution of each parameter. We extended the parameter space for the aerosol forcings beyond their feasible ranges to account for the weaker aerosol signals compared to the greenhouse gas forcings. Still, we found that many simulations are dominated by the response to greenhouse gas forcings. Future studies may wish to compare the emulator performance under different ratios of extended parameters to realistic parameters. Alternative designs, such as selecting some simulations with restricted greenhouse gas perturbations, which lead to stronger aerosol signals could potentially improve performance in this range. Furthermore, Bayesian optimization techniques could be used to directly identify new parameter choices for additional GCM simulations that would reduce emulator error or uncertainty (Shahriari et al., 2016)."**

3 Technical Corrections and Suggestions

• Lines 97-114 – This paragraph describes how the surface temperature 'response' is defined. It would be clearer to have this as a mathematical formula as the key quantity of interested to be emulated.

Lines 111-139:

> **We estimate the control surface temperature, by averaging over all years and all control simulations $\overline{T}^{CTRL}$.** All perturbation runs are initialized from the **same state and run for 5 years.** The well-mixed GHGs are perturbed via their global concentrations, while for the short-lived pollutants we use scaling factors for emissions over broad regions, described further in Section 2.2. **We define these as inputs to the GCM, denoted x^p for perturbation p. We estimate the perturbation surface temperatures by averaging over the 5 years and we denote this as $f_{GCM}(x^p)$.** We define the **short-term** surface temperature 'response' as
>
> $$y^p = f_{GCM}(x^p) - \overline{T}^{CTRL}$$
>
> The surface temperature response is defined at every grid cell of the GCM, giving a complete spatial map, **i.e., $y = (y_1, y_2, \cdots, y_N)$ for all $N = 27,648$ grid cells.**

• Line 112 – It is written that yGP ≈ y. This may be ambiguous and require a more

mathematically precise definition. For example, is it that the GP mean is a close approximation of y, or does this also encapsulate that the uncertainty is "small"?

We have removed this and improved our description of the GP (this directly follows on from paragraph pasted above):

Lines 140 – 145:

2.2 Gaussian process emulator

Our perturbation simulations create $P = 80$ input-output pairs which we call our training data $D = (x, y)$. We use a GP emulator to estimate the surface temperature response to a **new** emissions scenario, x^*. **The GP provides a probability distribution for each x^*, from which we can sample, i.e.,**

$$y_{GP} \sim GP(x^*|D)$$

This means we can also estimate the mean μ_{GP} and uncertainty σ_{GP}^2 for each x^*.

• Lines 211-212 – This is more commonly referred to as a maximin Latin hypercube design.

We add "maximin"

• Line 228 – The kernel K(x, x′) = Cov[f(x), f(x′)], thus G(x) and G(x′) should be replaced f(x) and f(x′) respectively.

Thank you for spotting this typo, we have corrected this.

• Line 243 – This is the first reference to a 40-year control run. Should this be introduced in section 2.1 where the six 5-year control runs are first described?

Yes, thank you, we added to Section 2.1 Global Climate Model Configuration:

Line 107-111:

"**The last 40-years of this simulation is treated as a control run** and this was then used as the starting point for simulations that serve as training data for the emulator.

Before computing any perturbation runs, **an additional** six 5-year-long control simulations were performed starting from this equilibrium state**, giving a total of 70-years of control simulations.**"

• *Line 360 – Point 1 – For the purpose of reproducibility, please provide more information on how the inputs are sampled including their variance (or standard deviation), and whether they are jointly or independently sampled.*

Line 449 "we sample all other inputs from **independent** normal distributions, **centered at present day levels with standard deviations set to ¼× the input range presented in Table 1**."

• *Line 360 – Point 2 – Ranges are defined for each of the input parameters. Are these limits enforced here?*

These limits are not strictly enforced here.

• *Line 511 – Add a citation to the heteroscedastic GP emulation paper Binois et al. (2018), and variance emulation, such as in Andrianakis et al. (2017).*

We have added these citations.

References

Andrianakis, Ioannis, Ian R. Vernon, Nicky McCreesh, Trevelyan J. McKinley, Jeremy E. Oakley, Rebecca N. Nsubuga, Michael Goldstein, and Richard G. White (2017). "History matching of a complex epidemiological model of human immunodeficiency virus transmission by using variance emulation". In: Journal of the Royal Statistical Society: Series C (Applied Statistics) 66.4, pp. 717–740. issn: 0035-9254. doi: 10. 1111/rssc.12198.
5
Binois, MickaÅNel, Robert B. Gramacy, and Mike Ludkovski (2018). "Practical Heteroscedastic Gaussian Process Modeling for Large Simulation Experiments". In: Journal of Computational and Graphical Statistics 27.4, pp. 808–821. doi: 10.1080/10618600. 2018 . 1458625. url: https : / / www . tandfonline . com / doi / full / 10 . 1080 / 10618600.2018.1458625.
Conti, Stefano and Anthony O'Hagan (2010). "Bayesian emulation of complex multioutput and dynamic computer models". In: Journal of Statistical Planning and Inference

140 (3), pp. 640–651. doi: 0.1016/j.jspi.2009.08.006.
Craig, Peter S., Michael Goldstein, Allan H. Seheult, and James A. Smith (1996). “Bayes linear strategies for matching hydrocarbon reservoir history”. In: Bayesian Statistics 5. Ed. by J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith. Proceedings of the Valencia International Meeting 5. Clarendon Press, pp. 69–95. isbn: 978-0198523567. doi: 10.1007/978-1-4612-2290-3_2.
Cressie, Noel A.C. (1993). Statistics for Spatial Data. Revised. Wiley Series in Probability and Statistics. New York, NY: John Wiley & Sons. isbn: 978-0-471-00255-0. doi: 10.1002/9781119115151.
Edwards, Tamsin L., Mark A. Brandon, Gael Durand, Neil R. Edwards, Nicholas R. Golledge, Philip B. Holden, Osabel J. Nias, Antony J. Payne, Catherine Ritz, and Andreas Wernecke (2019). “Revisiting Antarctic ice loss due to marine ice-cliff instability”. In: Nature 566, pp. 58–64. doi: 10.1038/s41586-019-0901-4.
Gneiting, Tilmann (2013). “Strictly and non-strictly positive definite functions on spheres”. In: Bernoulli 19.4, pp. 1327–1349. doi: 10.3150/12-BEJSP06.
Gu, Mengyang and James O. Berger (2016). “Parallel Partial Gaussian Process Emulation for Computer Models with Massive Output”. In: The Annals of Applied Statistics 10.3, pp. 1317–1347. issn: 1932-6157. doi: 10.1214/16-AOAS934.
Gu, Mengyang, Jesus Palomo, and James O. Berger (2019). “RobustGaSP: Robust Gaussian Stochastic Process Emulation in R”. In: The R Journal 11.1, pp. 112–136. doi: 10.32614/RJ-2019-011.
Higdon, Dave, James Gattiker, Biran Williams, and Maria Rightley (2008). “Computer Model Calibration Using High-Dimensional Output”. In: Journal of the American Statistical Association 103.482, pp. 570–583. doi: 10.1198/016214507000000888.
Johnson, J. S. et al. (2020). “Robust observational constraint of uncertain aerosol processes and emissions in a climate model and the effect on aerosol radiative forcing”. In: Atmospheric Chemistry and Physics 20.15, pp. 9491–9524. doi: 10.5194/acp-20-9491-2020. url: https://acp.copernicus.org/articles/20/9491/2020/.
Lee, L. A., K. S. Carslaw, K. J. Pringle, and G. W. Mann (2012). “Mapping the uncertainty in global CCN using emulation”. In: Atmospheric Chemistry and Physics 12.20, pp. 9739–9751. doi: 10. 5194/acp- 12 - 9739- 2012. url: https: //acp. copernicus.org/articles/12/9739/2012/.
Rougier, Jonathan (2008). “Efficient Emulators for Multivariate Deterministic Functions”. In: Journal of Computational and Graphical Statistics 17.4, pp. 827–843. doi: 10.1198/106186008X384032.

Rougier, Jonathan, Serge Guillas, Astrid Maute, and Arthur D. Richmond (2009). “Expert Knowledge and Multivariate Emulation: The Thermosphere–Ionosphere Electrodynamics

General Circulation Model (TIE-GCM)". In: Technometrics 51.4, pp. 414–424. doi: 10.1198/TECH.2009.07123.

Salter, James M. and Daniel B. Williamson (2022). "Efficient Calibration for High-Dimensional Computer Model Output Using Basis Methods". In: International Journal for Uncertainty Quantification 12.6, pp. 47–69. issn: 2152-5099. doi: 10.1615/Int.J.UncertaintyQuantification.2022039747.

Salter, James M., Daniel B. Williamson, John Scinocca, and Viatcheslav Kharin (2019). "Uncertainty Quantification for Computer Models With Spatial Output Using Calibration-Optimal Bases". In: Journal of the American Statistical Association 114.528, pp. 1800–1814. doi: 10.1080/01621459.2018.1514306.

Vernon, Ian, Michael Goldstein, and Richard G. Bower (2010). "Galaxy Formation: a Bayesian Uncertainty Analysis". In: Bayesian Analysis 5.4, pp. 619–670.

Wilkinson, Richard D. (2010). "Bayesian Calibration of Expensive Multivariate Computer Experiments". In: Large-Scale Inverse Problems and Quantification of Uncertainty. Ed. by Lorenz Biegler, George Biros, Omar Ghattas, Matthias Heinkenschloss, David Keyes, Bani Mallick, Youssef Marzouk, Luis Tenorio, Bart van BloemenWaanders, and Karen Willcox. Wiley. Chap. 10, pp. 195–215. isbn: 9780470697436. doi: 10.1002/9780470685853.ch10.