Geosci. Model Dev., 6, 1715–1728, 2013
www.geosci-model-dev.net/6/1715/2013/
doi:10.5194/gmd-6-1715-2013

Geoscientific
Model Development
Open Access

The potential of an observational data set for calibration of a computationally expensive computer model

D. J. McNeall[1], **P. G. Challenor**[2], **J. R. Gattiker**[3], **and E. J. Stone**[4]

[1]Met Office Hadley Centre, Exeter, UK
[2]College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, UK
[3]Los Alamos National Laboratory, Los Alamos, New Mexico, NM 87545, USA
[4]School of Geographical Sciences, University of Bristol, Bristol, UK

Correspondence to: D. J. McNeall (doug.mcneall@metoffice.gov.uk)

Received: 15 February 2013 – Published in Geosci. Model Dev. Discuss.: 9 April 2013
Revised: 30 July 2013 – Accepted: 5 September 2013 – Published: 21 October 2013

Abstract. We measure the potential of an observational data set to constrain a set of inputs to a complex and computationally expensive computer model. We use each member in turn of an ensemble of output from a computationally expensive model, corresponding to an observable part of a modelled system, as a proxy for an observational data set. We argue that, given some assumptions, our ability to constrain uncertain parameter inputs to a model using its own output as data, provides a maximum bound for our ability to constrain the model inputs using observations of the real system.

The ensemble provides a set of known parameter input and model output pairs, which we use to build a computationally efficient statistical proxy for the full computer model, termed an emulator. We use the emulator to find and rule out "implausible" values for the inputs of held-out ensemble members, given the computer model output. As we know the true values of the inputs for the ensemble, we can compare our constraint of the model inputs with the true value of the input for any ensemble member. Measures of the quality of constraint have the potential to inform strategy for data collection campaigns, before any real-world data is collected, as well as acting as an effective sensitivity analysis.

We use an ensemble of the ice sheet model Glimmer to demonstrate our measures of quality of constraint. The ensemble has 250 model runs with 5 uncertain input parameters, and an output variable representing the pattern of the thickness of ice over Greenland. We have an observation of historical ice sheet thickness that directly matches the output variable, and offers an opportunity to constrain the model. We show that different ways of summarising our output variable (ice volume, ice surface area and maximum ice thickness) offer different potential constraints on individual input parameters. We show that combining the observational data gives increased power to constrain the model. We investigate the impact of uncertainty in observations or in model biases on our measures, showing that even a modest uncertainty can seriously degrade the potential of the observational data to constrain the model.

1 Introduction

Computer models (referred to hereon as computer simulators) are used in a wide variety of computer experiments, for the understanding and prediction of real-world systems (see e.g. Santner et al., 2003 for examples). Such simulators contain uncertain parameters that may represent real but unknown physical constants, or be artefacts of the simplification (and therefore parameterization) of complex physical processes. It is important to choose an appropriate set of parameters with which to run the simulator, in order that simulations match the behaviour of the true system as closely as possible.

This raises questions: What observational data might we collect in order to effectively match the simulator to the system under study? And how valuable might they be in constraining our input parameters? We imagine a situation where a new observational campaign of the system under study is being considered, and there is a substantial cost associated with making new observations of the system. We might

Published by Copernicus Publications on behalf of the European Geosciences Union.

extend this to cases where there are observations, but we could reduce their uncertainty. Finally, we might have a limited budget, which we can choose to spend on reducing observational uncertainty, or on improving the simulator – in effect, reducing the simulator discrepancy, or its associated uncertainty. To guide the observational campaign, we would like to know the potential of an observation, with a particular uncertainty, to constrain our simulator *before* we make the observation.

The comparison of simulators with observations from the appropriate real-world system, in order to choose a set of appropriate parameters, is known as calibration. This paper introduces a method for estimating the potential of a data set for calibrating a simulator, when that simulator is computationally too expensive for brute force methods of calibration or tuning to be effective. We use an ensemble of the simulator output as a synthetic data set, treating output from an ensemble member as if it were an observation of the real system under study. We propose that our ability to calibrate the simulator when we know the true set of parameters (as in our ensemble), gives us a theoretical upper limit on our ability to calibrate the simulator in the real system.

In this synthetic test bed, we can examine the impact on the calibration of adding observational uncertainty, or simulator discrepancy uncertainty. We advise caution, as the true simulator discrepancy remains unknown, and might be different from anything that we can reasonably simulate. However, we believe that our metrics give a good guide to the maximum constraint possible, given a particular simulator, statistical framework, and data set.

Once a simulator is calibrated, it can be run to predict the behaviour of the system under untested circumstances. For example, climate simulators calibrated to historical data can be used to project and constrain the behaviour of the Earth system in the future under various greenhouse gas emission scenarios (Sexton et al., 2012; Sexton and Murphy, 2012; Rougier, 2007; Tebaldi and Knutti, 2007). Such simulators are often computationally expensive to run, such that there are usually only a small set of runs of the code with which to estimate a potentially large number of these uncertain but tuneable parameters within the simulator.

A probabilistic calibration allows for uncertainty in observational data, and for the fact that the simulator does not perfectly represent the true system. Such probabilistic calibration allows a range for each of the input parameters, assigning a probability that each of the input parameters in a set might best match the simulator to the true system. In this case, a probabilistic prediction can be made by weighting the prediction of the simulator according to the probability of the corresponding set of input parameters being correct.

Metrics for the potential of data to constrain input parameters have been proposed when working with computationally cheap simulators and probabilistic calibration methods; for example to simulate atmospheric aerosols (Partridge et al., 2012), or terrestrial ecosystem models (Ziehn et al., 2012). Here, we extend the methods for calculating these kinds of metrics to computationally expensive simulators.

Calibration of a computationally expensive simulator can be efficiently achieved using an emulator: a fast and computationally cheap statistical proxy for the full simulator. Use of an emulator for calibration in a Bayesian setting was pioneered by Kennedy and O'Hagan (2001), with Wilkinson (2011) offering a review of recent developments. An alternative approach, also using emulation techniques, is the history matching of Craig et al. (1996, 1997, 2001). History matching places more emphasis on ruling out parameter sets where the simulator performs poorly, whereas probabilistic calibration tends to down-weight poorly performing parameter sets. While these approaches differ in their interpretations of the meaning of the simulator, both share a notion of distance of simulator output from observations of the real system, as a measure of simulator quality.

Our metrics can also be viewed as a form of global sensitivity analysis (Saltelli et al., 2000). Sensitivity analysis (SA) in this context is concerned with quantifying the strength of the relationship between the inputs and outputs of a simulator. This relationship is often couched in terms of the induced change in simulator output, for a given change in simulator input. We are interested in inverting this measure, and finding the implied uncertainty of a simulator input, given an output. Trivially, if the output of a simulator is not sensitive to an input, then the data corresponding to the output will not have the power to constrain the input parameters. In addition, even where there may be a unique forward mapping from inputs to outputs of a simulator, this is not necessarily true of the inverse mapping. A single output may have many corresponding inputs. An approach to probabilistic SA for expensive computer simulators is introduced by Oakley and O'Hagan (2004). Our approach draws on those techniques, particularly in the use of a Gaussian process emulator as a proxy for the computer simulator.

We first briefly introduce history matching as a method of solving inverse problems in the context of computer simulators in Sect. 2.1. We then introduce some empirical metrics for the ability of an observation to constrain the simulator input parameters in Sect. 2.2. In Sect. 2.3, we introduce emulators, and explain how they might be used in calculating the metrics introduced in the previous section. We apply our methods of constraint to an ensemble of a computationally expensive ice sheet simulator, and show that they work in Sect. 3. We introduce the results in Sect. 3.2, and discuss them and their implications for future research directions in Sect. 4. Finally, we offer some conclusions in Sect. 5.

2 Methods

2.1 Solving the inverse problem

We would like a metric for the strength of an observation of a system to calibrate (to constrain, or find good values for) a set of uncertain parameters in our computer simulator of that system. This equates to asking "how well can we solve the inverse problem, of estimating the parameters of a simulator, given some data?". There are at least two approaches to solving the inverse problem: probabilistic calibration, and history-matching techniques.

In a probabilistic calibration, a probability is assigned to a candidate set of inputs, depending on how well the corresponding output of the simulator matches observations, and the prior probability (before any data is seen) of the candidate point being "correct" in some manner.

Following Rougier (2007), we represent a particular set of d input parameters as vector $\boldsymbol{x} = x_1 \ldots x_d$, set within ($\in$) a "parameter" or "input" space $\mathcal{X}$, judged to be plausible by the modeller, before the simulator is run. We assume that this plausible space corresponds to a "prior" probability distribution, if we were to carry out a fully Bayesian analysis. Similarly, we represent the simulator output as $\boldsymbol{y} \in \mathcal{Y}$, representing the state of some physical aspect of the system. We represent the simulator as a deterministic function $g(.)$, so that when run at a particular input parameter set $\boldsymbol{x}$, it always returns the same value of $\boldsymbol{y}$. The simulator is complex enough that we cannot trivially predict the output $\boldsymbol{y}$ at a given $\boldsymbol{x}$ before the simulator is run. We can represent output $\boldsymbol{y}$ as an uncertain function of input $\boldsymbol{x}$ thus:

$$\boldsymbol{y} = g(\boldsymbol{x}) . \tag{1}$$

The relationship between the simulator output $\boldsymbol{y}$ and an observation z of the real system is represented by the equation

$$z = g(\boldsymbol{x}^*) + \delta(\boldsymbol{x}^*) + \boldsymbol{e} , \tag{2}$$

where $\boldsymbol{e}$ represents measurement errors in the observations, and δ is the simulator discrepancy; the difference between the real system, and the simulator when run at its "best" input, $\boldsymbol{x}^*$. This best input is therefore defined as the point which minimises the difference between the observations and the simulator output, given any known systematic errors (biases) in simulator discrepancy or in observations.

In calibrating the simulator, we compare a set of observations of the true system, z, with the corresponding representative output of the simulator $\boldsymbol{y}$, and through the mapping in Eq. (1) we find a set of input parameters that is, by some measure (but not necessarily all measures), good. In general, we assume that parameter sets which represent the real system well produce a smaller difference between simulator outputs $\boldsymbol{y}$ and observations z, than do poor choices of inputs, and have a corresponding higher probability of representing the best input. In addition, we assume that there are places within $\mathcal{X}$ where it is possible to run the simulator, that nevertheless we judge as not well representing the true system being modelled. We would like to exclude these regions from our analysis as "implausible", in effect setting their probability to zero. Constraining $\mathcal{Y}$ to a smaller representative region by comparing it with observations z therefore implies a constraint on $\mathcal{X}$.

This constraint might be achieved through a fully probabilistic calibration, simultaneously estimating probability distributions for $\boldsymbol{x}^*$, and for simulator discrepancy δ, as in Kennedy and O'Hagan (2001). We use an alternative history-matching approach, based on the concept of implausibility, introduced by Craig et al. (1996). A full description of the benefits of history matching for expensive simulations can be found in Vernon et al. (2010). Briefly, the aim is to rule out as implausible, sets of parameters space where the simulator is a very poor fit to observations of the real system. Any set that is "not ruled out yet" is passed to further analysis. The implausibility measure must take into account (a) the fact that the observations are uncertain, (b) that we have uncertainty about ways in which the simulator might be wrong (the discrepancy), and (c) that we do not fully know the simulator behaviour, due to our limited ability to run the simulator.

We use an implausibility measure I that takes all of these uncertainties into account, writing

$$I^2 = \frac{|E\left[g(\boldsymbol{x})\right] - z|^2}{\operatorname{Var}\left[g(\boldsymbol{x}) + \delta(\boldsymbol{x}) + \boldsymbol{e}\right]} . \tag{3}$$

An input is more implausible, the further the corresponding output lies from observations of the true system. However, if the observations, the simulator output at that input, or the simulator discrepancy are more uncertain, that same input would be less implausible.

We regard any point where implausibility is below a threshold value of 3 as "not implausible", and accept it as a candidate for the best input. This threshold comes from the 3σ rule of Pukelsheim (1994), which states that for unimodal distributions, if $\boldsymbol{x} = \boldsymbol{x}^*$, then $I < 3$, with a probability greater than 0.95. This holds true even for highly skewed, or heavy-tailed distributions.

In this framework, comparing the simulator with more than one type of observation is simple. In the case where different types of observation imply different implausibility, we take the maximum implausibility at the candidate input point x. This allows for progressive ruling out of parameter space, as more observations become available. A multivariate alternative to the maximum implausibility measure is introduced by Vernon et al. (2010), along with modifications that make the maximum implausibility measure less sensitive to inaccuracies in an individual emulator. The high accuracy of the emulator used in this study means that we can use the simplest method.

2.2 Metrics of constraint

We would like to assign a score or metric for the ability of a particular observation z of the real system to constrain our choice of a good set of input parameters within $\mathcal{X}$. There are a number of ways that we might measure this, limited by some practical considerations. We propose two primary metrics: (1) the marginal range of plausible space in each input dimension, relative to the initial estimate and (2) the volume of "not implausible" input space, relative to the initial estimate.

2.2.1 Marginal range of "not implausible" input space

The marginal range R of an individual input is the largest range for each input parameter that we can find where $I < 3$. This is measured relative to the marginal ranges considered plausible before the simulator was run. While this measure can be useful as a simple sensitivity analysis, it should be treated with caution, and we regard it as inferior to the *volume* of "not implausible" space metric, outlined in the next section. The *range* measures only the one-dimensional projection of the "not implausible" input space. The true range of an individual might be very much smaller (and the metric correspondingly more useful), if we were to gain information about another input parameter, for example.

2.2.2 Volume of "not implausible input" space

We can define a volume V of "not implausible" input parameter space, or alternatively that input space "not ruled out yet" – as the region where $I < 3$. We can estimate the relative volume of this space, with a Monte Carlo sample from the initially plausible space $\mathcal{X}$. Using an indicator function $\mathcal{I}(I < 3)$, where $\mathcal{I} = 1$ if true, and 0 if not, we take n samples from $\mathcal{X}$, and estimate the volume as

$$V = \frac{1}{n}\sum_{i=1}^{n}\mathcal{I}(I < 3)\,. \tag{4}$$

We must be careful to take enough samples to ensure that this estimate is accurate, as well as taking into account the sometimes counter-intuitive nature of high dimensional space. For example, an observation that constrains the plausible volume to half the range of each input in a 5-dimensional input space would have reduced the space to $0.5^5 \approx 3\,\%$ of its original volume. However, such a reduction in volume can be achieved by constraining a single input to 3 % of its original range, with no constraint on any other input.

2.3 An emulator for computationally expensive simulators

We are concerned with the case where the simulator is computationally expensive, and complex enough that we cannot trivially predict the output of the simulator before we run it. We therefore cannot run the simulator enough times to comprehensively explore the mapping of $\mathcal{X}$ to $\mathcal{Y}$. We could, for example, run a collection of simulations in an optimisation routine to find $\boldsymbol{x}^*$. This is unlikely to be a practical solution, given the possibly complex nature of $\boldsymbol{z}$, the difficulty of searching high dimensional spaces which can have many local minima, and conflicting demands on expensive simulator output.

A more flexible solution is to run the simulator at a carefully designed collection of points in $\mathbf{X} \in \mathcal{X}$, with associated output $\mathbf{Y}$, called an ensemble, and use this to build a statistical model to predict the output $\boldsymbol{y}$, at untested points within $\mathcal{X}$. This statistical model, termed an emulator, is computationally cheap and fast to run, and therefore can replace our simulator in any analysis of the ensemble. The emulator returns an estimated probability distribution for simulator output given an input.

It is important to design the ensemble well, in order to build a good emulator. The simulator should provide good coverage of the input parameter space, in order that interactions between parameters might be well estimated. It should also span enough parameter space that the emulator is not called to extrapolate far beyond the design points, or parameter values where the emulator has been validated. A good option is the Latin hypercube design of McKay et al. (1979), and its space-filling variants.

The emulator, denoted $\eta(.)$, provides us with a complete mapping of $\mathcal{X}$ to $\mathcal{Y}$, with some uncertainty. If this uncertainty is tolerably small, we can use the emulated best estimate of simulated output in any analysis where we would normally use the simulator directly. We denote the best estimate for $\boldsymbol{y}$ at any given $\boldsymbol{x}$ as $\hat{\boldsymbol{y}} = \eta(\boldsymbol{x})$.

2.4 Using an ensemble to find an upper bound of potential constraint

With an ensemble of a priori plausible simulator evaluations, we let the simulator output $\boldsymbol{y}$ take the place of a theoretical observational data set $\boldsymbol{z}$ in our analysis. We estimate "not implausible" candidates for $\boldsymbol{x}^*$ for a given ensemble member, given its output $\boldsymbol{y}$. The candidates will span a region within the original input parameter space. We can calculate the metrics of constraint for that region, introduced in Sect. 2.2, and also check that the true value of $\boldsymbol{x}^*$ falls within the "not implausible" region.

For computational efficiency, we let the emulator $\eta(.)$ take the place of our simulator $g(.)$. Simulator discrepancy $\delta(.)$ and observational error $\boldsymbol{e}$ (along with their respective uncertainties) are both zero in this setting, as we know the observational data perfectly and we are using the same simulator across the ensemble. We can easily add in a simulator discrepancy or observational error of our choice, in order to test their impact on our ability to constrain $\boldsymbol{x}^*$.

We use a leave-one-out cross-validation (LOOCV) style test on the ensemble, to find metrics of constraint at a sample across input space. For each ensemble member in turn $i = 1 \dots n$ we treat the output $\boldsymbol{y}_i$ from an ensemble member as an observation of the true system. We build an emulator, conditioned on the entire ensemble, except input $\boldsymbol{x}_i$ and output $\boldsymbol{y}_i$. By finding the "not implausible" region, where our implausibility measure $I < 3$ for each output in the ensemble $\boldsymbol{y}_1 \dots \boldsymbol{y}_n$, we can obtain a sample of possible constraints that an observation would give, if it were found to be $\boldsymbol{y}_i$.

We take a large Monte Carlo sample from the prior distribution at a large number of points within $\mathcal{X}$, and use the emulator to predict $\hat{\boldsymbol{y}}_i$ at each candidate point. We then find the implausibility I at each emulated input, given the uncertainty about the true value of the simulator at that point, provided by the emulator. We calculate the metrics of plausible marginal input parameter range R, and plausible input space volume V, using the emulated implausibility for each point.

Repeating this process across the ensemble, we obtain a sample of n of each of the constraint metrics $V_1 \dots V_n$, and $R_1 \dots R_n$, where we have n ensemble members. Each sample represents what the constraint might be if the true observation were to fall at $\boldsymbol{y}_i$, so we see that there is some uncertainty in the ability of the data to constrain the inputs, depending upon where in the ensemble the true data might fall.

It is important that the ensemble output spans a range wide enough to encompass any reasonable combination of observation, simulator discrepancy and observational uncertainty. This is to avoid the situation where (for example) the observation falls well outside the range of simulated output, and all of the input space is effectively ruled out immediately. In this situation, the analysis would be iterated, with new judgements about the uncertainty of simulator discrepancy.

3 An example using an ice sheet simulator

We investigate the utility of an emulator/observational data set combination, for the calibration of the ice sheet simulator Glimmer (version 1.04) (Rutt et al., 2009; Payne, 1999). We have access to an ensemble of 250 simulator runs, with 5 uncertain inputs, and an output variable, ice thickness, at each point in a 76 × 141 grid covering the Greenland Ice Sheet (GrIS). This ensemble was generated and examined in Stone et al. (2010); details of the inputs and outputs are summarised in Table 1. The simulator is sufficiently computationally expensive to serve as a test bed for our methods, while being relatively straightforward to run in an ensemble of several hundred members.

The ensemble input points are sampled from independent uniform distributions of simulator inputs, using a Latin hypercube sampling strategy. We normalize all inputs to a zero-one scale, based on the expert-elicited limits of the ensemble design.

The simulator output domain matches real-world observations of ice sheet thickness (Bamber et al., 2001) interpolated to the simulator grid (Fig. 1), and shown here to aid interpretation of the data. We can summarise the output variable ice thickness, into a univariate output, in three ways. First, we can find ice volume (denoted ICEVOL), by summing the ice thickness over the entire simulator domain. Second, we can take the surface area (ICESA) of the ice sheet. Third, we can examine the maximum thickness (MAXTHK) of the ice sheet. It is important to simulate all of these variables correctly, in order to have confidence that our ice sheet simulator is capturing the relevant dynamics of the GrIS. In Fig. 2, we plot the marginal relationships between each pair of inputs and outputs. We see that, even though the summary outputs are from the same field variable, the output summaries are affected by input dimensions in different ways. Again, simulator outputs are normalized to a zero-one scale.

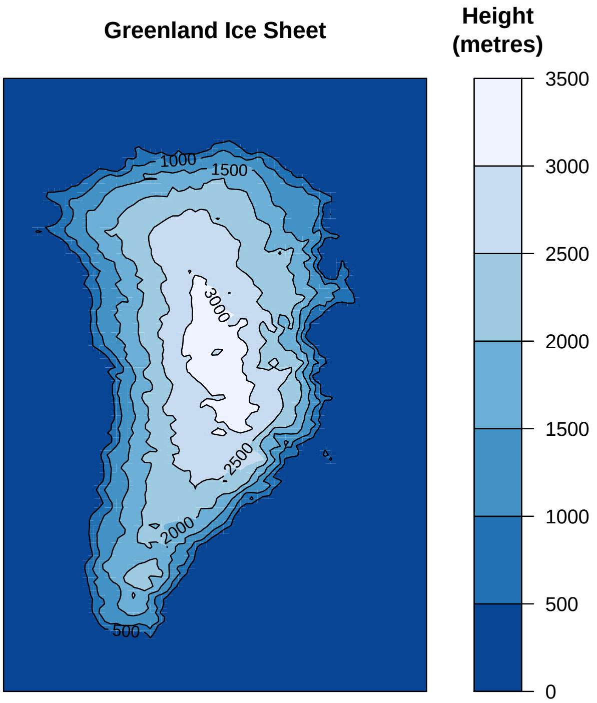

Fig. 1. Observations of ice sheet thickness over the Glimmer domain, from Bamber et al. (2001).

3.1 Building and checking the emulator

A first task is to build an emulator that we are confident accurately represents the forward mapping between input and output space. We use a Gaussian process emulator, implemented in the package BACCO (Hankin, 2005), using the statistical software R (R Core Team, 2012). The emulator is composed of a basic linear statistical model, along with a more flexible part known as a Gaussian process, conditional on a set of roughness parameters. There is one roughness parameter for each simulator input-output relationship. The roughness parameters represent the length scales in each input dimension at which a simulator output becomes uninformative about

Table 1. Expert-elicited ranges for input parameters of Glimmer, and the corresponding ranges of the output parameters.

	Units	Abbrev.	Min	Max
Input Parameter				
Positive degree day factor for snow	$\mathrm{mm\,d^{-1}\,°C^{-1}}$	PDDFS	3	5
Positive degree day factor for ice	$\mathrm{mm\,d^{-1}\,°C^{-1}}$	PDDFI	8	20
Near-surface lapse rate	$\mathrm{°C\,km^{-1}}$	NSLR	−8.2	−4
Flow enhancement factor	–	FEF	1	5
Geothermal heat flux	$\mathrm{mW\,m^{-2}}$	GHF	−61	−38
Output Parameter				
Ice volume	$\mathrm{m^3}$	ICEVOL	3.1×10^6	4.3×10^6
Ice surface area	$\mathrm{m^2}$	ICESA	2.0×10^6	2.4×10^6
Maximum ice thickness	m	MAXTHK	3.0×10^3	3.7×10^6

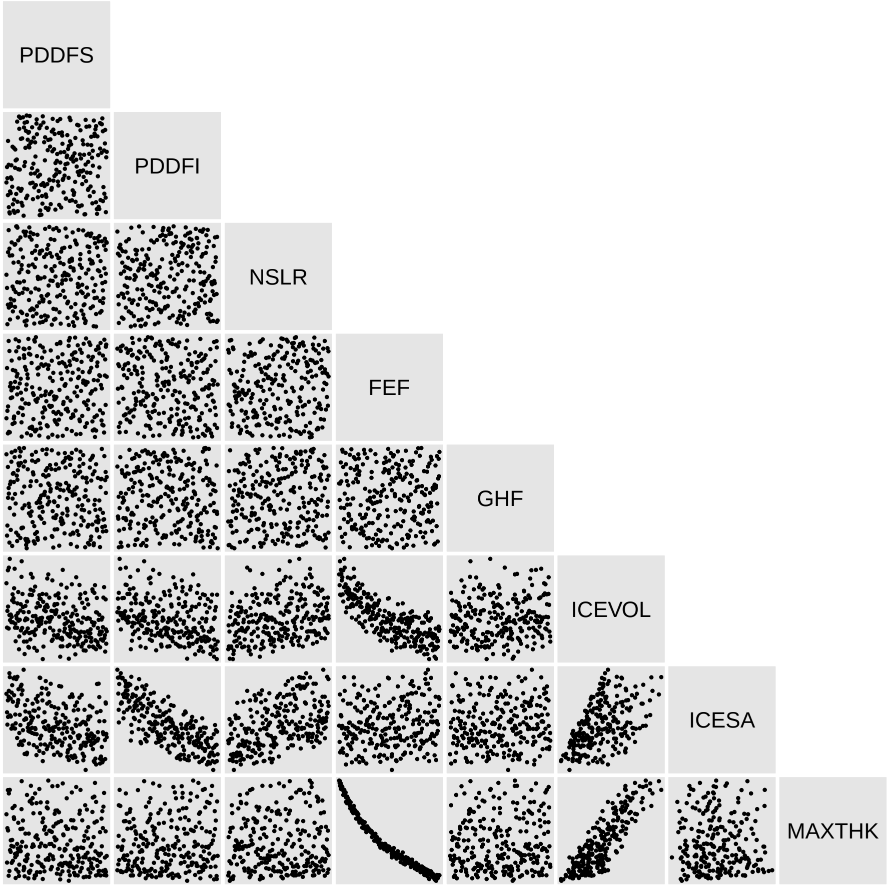

Fig. 2. Summary pairs plot of relationships between simulator inputs and outputs. All inputs and outputs are normalised to a zero-one scale, relative to the limits of the ensemble.

a nearby simulator output. If the simulator is very rough in a dimension, a simulator run will contain little information about a nearby run, and uncertainty will increase rapidly beyond any known simulator run. We use a single set of roughness parameters, estimated empirically from the entire ensemble. It would be possible to estimate the roughness parameters for each leave-one-out subset of data, but we find that in practice this makes very little difference to the results at markedly increased computational cost. The parameters

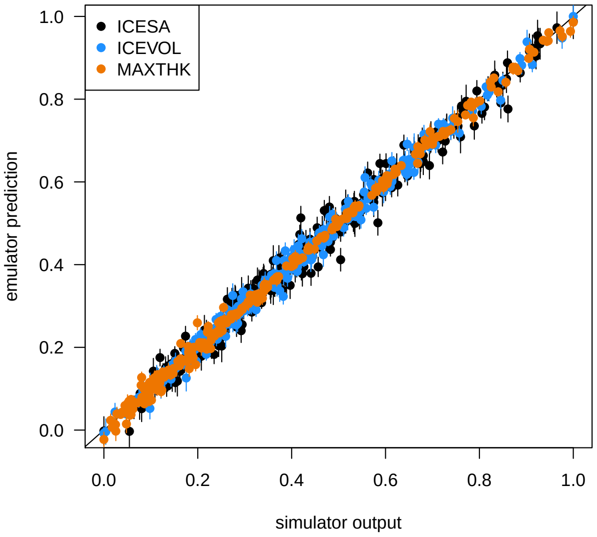

Fig. 3. Leave-one-out cross-validation for a Gaussian process emulator, showing good performance in prediction. We exclude a member from the ensemble, and predict the output, given the set of input parameters. We repeat this process across the ensemble, for three summary simulator outputs. Vertical lines represent ±1 standard deviation.

are estimated via the posterior mode, as set out by Oakley (1999).

The emulator fits a “best estimate” of the simulator output at a particular input, smoothly through each of the available outputs. It then estimates the uncertainty at each point, with the uncertainty at known simulator runs reducing to zero, and growing with distance from each known point. There is no “nugget” term, and so the emulator is constrained to fit the points where the simulator has been run exactly. We build a separate emulator for each output, individually. Mathematical details of the GP emulator can be found in, e.g. Kennedy and O’Hagan (2001), or Oakley and O’Hagan (2004).

We check the performance of the emulator by performing both a forward, and an inverse leave-one-out cross-validation analysis. Each ensemble member is excluded in turn, and the emulator built on the remaining members of the ensemble. First, we exclude simulator output, and predict it using the most likely value of the emulator uncertainty distribution, given the set of inputs. The prediction plots for the three output summaries given in Fig. 3, show that the emulator works well, with small error and no detectable biases, across the ensemble, and for each of the three outputs. Second, we find the implausibility I of the true held-out input, given the simulator output. We find this to be below the threshold of 3 in all but 3, 1 and 2 ensemble members, for ICEVOL, ICESA and MAXTHK, respectively. In these members, the value of I is always below 4.

3.2 Results

To demonstrate our methods, we first show an example of constraining input parameter space of the (arbitrarily chosen) first ensemble member, with no additional observational or simulator discrepancy uncertainty. We show two ways of visualising the constraint of input space in Figs. 4 and 5. After sampling uniformly from the entire input space, we plot two-dimensional projections of those emulated input points assigned “not implausible” by our method, when we use all three data summaries to constrain the inputs (Fig. 4). It is clear that the true value of the inputs (green point) lies within the region defined by the two dimensional projections. A similar result is obtained looking at the parallel coordinates plot (Fig. 5), showing the full location of the “not implausible” emulated ensemble members (red), along with the target ensemble member (blue). Again, those points calculated as “implausible” are excluded from the plot. It is possible to clearly see how well the input parameter FEF is constrained, using the ensemble data. As each input is plotted over its entire range, it is easy to see the “not implausible” range of each parameter in Fig. 5, as the difference between the uppermost and lowermost points on each axis.

Once we have established that the emulator is accurate to an acceptable degree, its flexibility allows us to conduct many useful analyses that are too expensive to conduct with the original simulator. For example, we can begin to study the behaviour of the simulator, in terms of its individual inputs. We conduct a “two-at-a-time” sensitivity analysis, and plot the results in Fig. 6. Again, we use the first ensemble member as an example. Each subplot shows the estimated implausibility measure I, when the named inputs are varied across a regular grid, and the remaining three inputs are held at their true values. The contributions to the final “maximum implausibility” measure from each observation type are shown in the inset (top right), and the true values of the ensemble member are plotted as a green point. In this kind of analysis, it quickly becomes clear that “not implausible” regions of input space often form hyperplanes within high dimensional input space.

We use our emulated implausibility method outlined in Sect. 2.4 in order to invert the emulator, and provide a set of metrics for the ensemble members in a leave-one-out fashion. We assume that we have no prior information on the precise location of any ensemble member within the input space, and so we use a uniform distribution across the ensemble as a prior distribution. We take a large Monte Carlo sample of inputs and corresponding outputs (order thousands) from the emulator over the entire domain, and find their implausibility I, according to Eq. (3). Using the emulated implausibility, we calculate V and R for each ensemble member.

We report results here for two situations. First, we neglect any observational or discrepancy uncertainty, and find the maximum possible constraint for a given data set/emulator pair. Second, we include a representative observational

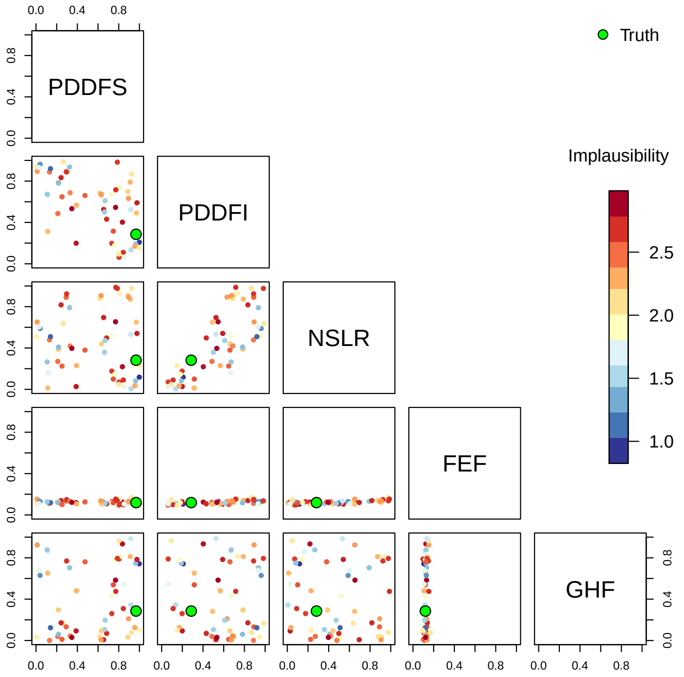

Fig. 4. Two-dimensional projections of emulated "not implausible" ($I < 3$) ensemble members, when the true inputs are those of the (arbitrarily chosen) first ensemble member. Implausibility is calculated as the maximum of that from all three summaries of the output data – ICEVOL, ICESA and MAXTHK. Emulated implausible members (not shown) are spread evenly through the input space. The true value of the inputs (the target) is shown as a green point.

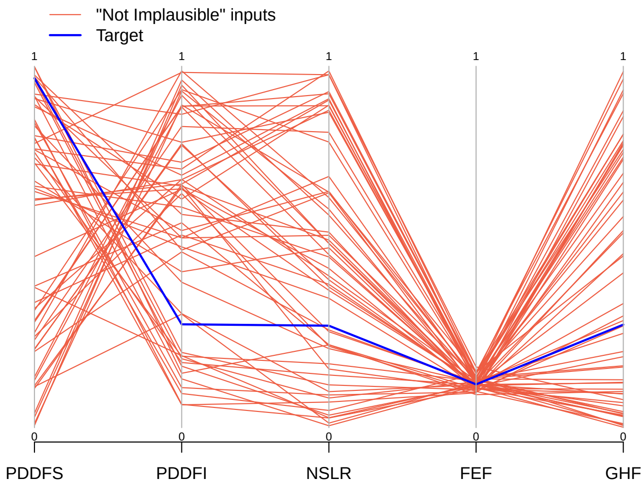

Fig. 5. Parallel coordinates plot of emulated "not implausible" ($I < 3$) ensemble members (red), when the true inputs are those of the (arbitrarily chosen) first ensemble member (blue). Lines join points on the y axis, normalised to the ensemble maxima and minima, with each line representing a point in parameter space. Implausibility is calculated as the maximum of all three summaries of the output data – ICEVOL, ICESA and MAXTHK. Emulated implausible members (not shown) are spread evenly through the input space, and would cover the entire range if shown.

uncertainty, in order to test the sensitivity of our metrics to a real-world situation. We fix the standard deviation of the representative observational error as 10 % of the maximum simulated value for each of the outputs in the ensemble. This uncertainty might also represent a discrepancy uncertainty, as observational and discrepancy uncertainty are added in the denominator in Eq. (3). We test each of our simulator outputs in turn, to find which might provide the most useful constraint overall, or for any of the particular simulator inputs.

In Fig. 7, we see the distribution across the ensemble of the constraint R – the range of inputs for each input parameter that are not implausible. The constraint for each parameter is represented by the block of colour, reaching a height on the y axis. An ensemble member filling the full height is marginally unconstrained by the data; a member reaching halfway up the y axis is constrained to 50 % of the range of the original ensemble. The ensemble members are plotted along the x axis, ordered from the strongest constrained member to the weakest, independently for each parameter.

Columns of individual plots show the results when summarising the simulator outputs in the three different ways, with the final column representing constraint combining all three ways of summarising the data. The top row, (marked a), represents the upper bound of constraint possible – that with no observational or simulator discrepancy uncertainty. The

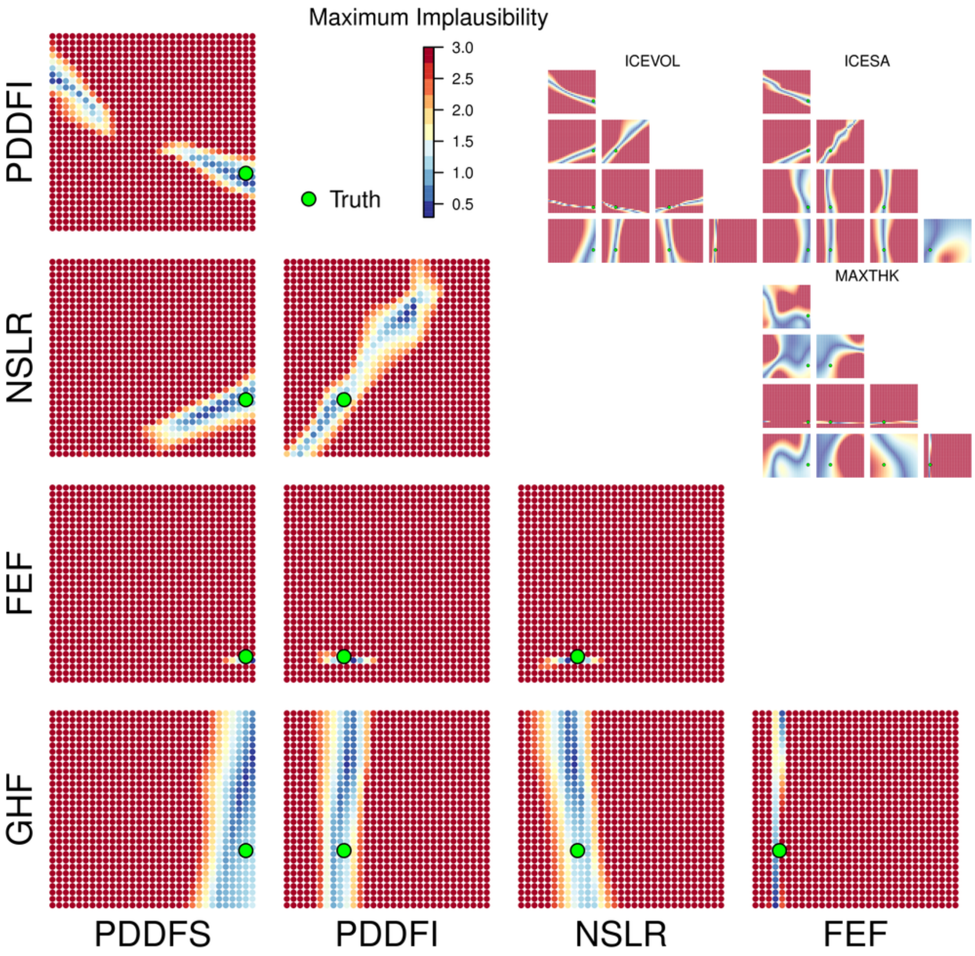

Fig. 6. A "two-at-a-time" sensitivity analysis of the (arbitrarily chosen) first ensemble member. Each subplot (main figure) shows the emulated implausibility measure I, when the named inputs are varied across a regular grid, and the remaining three inputs are held at their true values. Contributions to the final "maximum implausibility" measure from each observation type are shown in the inset (top right), and the true values of the ensemble member are plotted as a green point.

lower row (marked b) shows the constraint with a simulated 10 % observational uncertainty.

These plots are summarised in Table 2, expressed as a percentage of the range of each input, which is on average ruled out as implausible when using a particular output for constraint. While this summary table is useful, it does not adequately describe the detail across the ensemble. For example, there is a large range of possible constraints if using ice surface area as a calibrating data set, even when we do not include observational or simulator discrepancy uncertainty (Fig. 9a). The input PDDFI might be constrained by up to 65 %, by this data set, or not at all, depending on where in input space the true input lies. In contrast, using maximum ice thickness (MAXTHK) as a calibrating data set will only have a significant constraining effect on input FEF. However, the smallest constraint observed is around 60 %, and largest is near 100 %.

The histograms in Fig. 8 represent the sample of the volume V of input space retained as "not implausible" across the ensemble. First, we focus on the upper bound constraint, with no additional observational uncertainty. Overall, we see that maximum ice thickness (MAXTHK), provides the strongest potential constraint of any individual data set, with all of the ensemble members being constrained to a volume of input space 13 % of the original volume. This is followed by ice volume, ICEVOL with all below 17 %, and then ice surface area ICESA, with all below 27 %. Combining the data, and rejecting an input with an implausibility $I > 3$ in any of the data, leads to a much stronger constraint, with all inputs parameters constrained to smaller than 4 % of the original volume of input space.

The impact of adding a representative observational error of 10 % of the ensemble maximum is considerable. We see in Fig. 8 that the overall ability of the data to constrain the

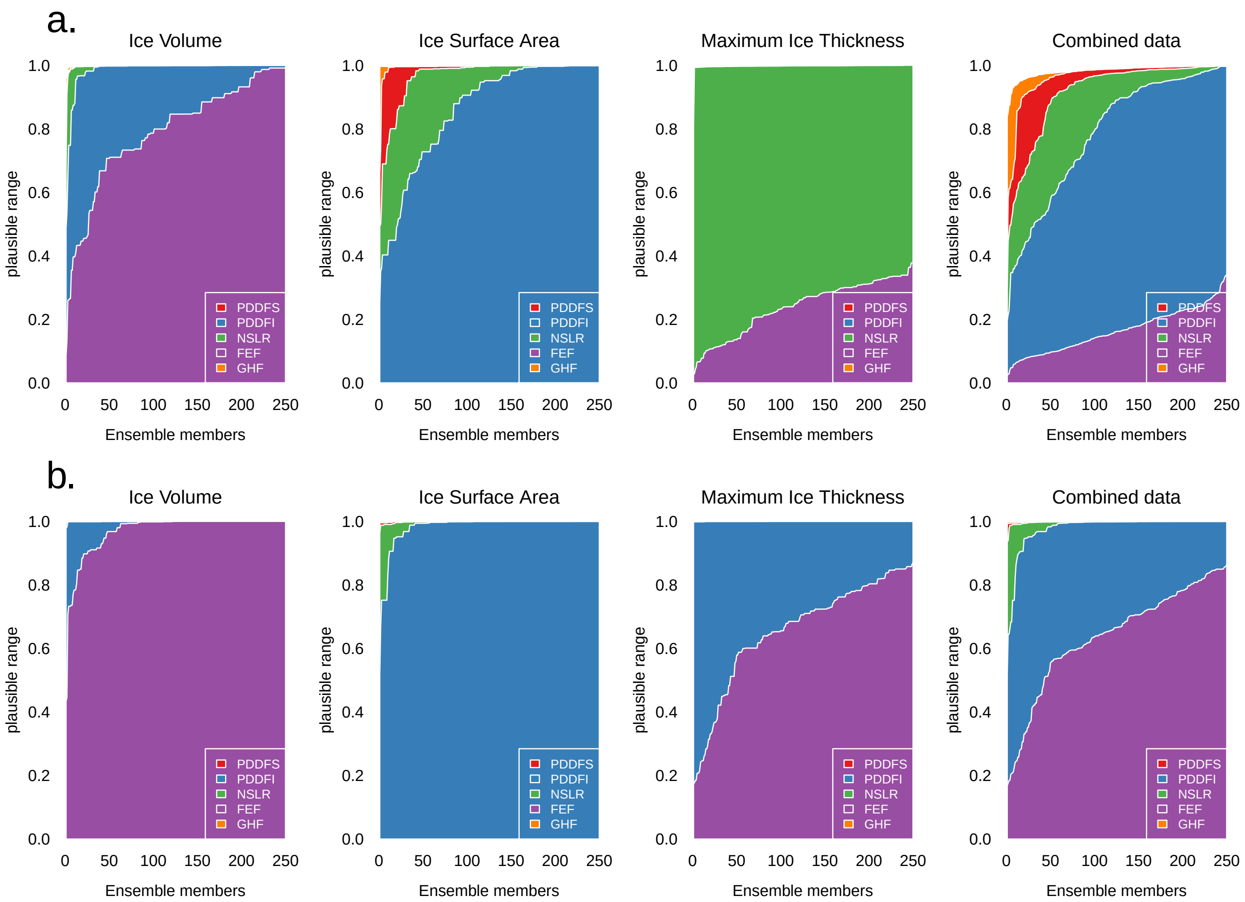

Fig. 7. The "not implausible" range of each input, R **(a)** assuming no observational or simulator discrepancy uncertainty and **(b)** with 10 % of maximum observational uncertainty (1 standard deviation). The "not implausible" range R across the ensemble is measured on the y axis of each plot, where different colours represent the input parameters. R varies according to ensemble member, with the ensemble members sorted from lowest to highest R, separately for each parameter. Individual plots represent the constraint using a single data summary, with the final plot in each row representing the constraint using the maximum implausibility of all three data summaries combined. Plot colours are ordered by the mean constraint of each parameter.

Table 2. The typical range of each marginal input range (%) ruled out as "implausible", when using a particular simulator output. The number is the mean of the implausible range, taken across the ensemble. "COMBINED" indicates the constraint when the maximum implausibility from all data streams is used.

	PDDFS	PDDFI	NSLR	FEF	GHF
No observational error					
ICEVOL	0	1	0	21	0
ICESA	1	14	3	0	0
MAXTHK	0	0	0	76	0
COMBINED	4	21	8	84	2
10 % observational error					
ICEVOL	0	0	0	3	0
ICESA	0	2	0	0	0
MAXTHK	0	0	0	34	0
COMBINED	0	2	0	36	0

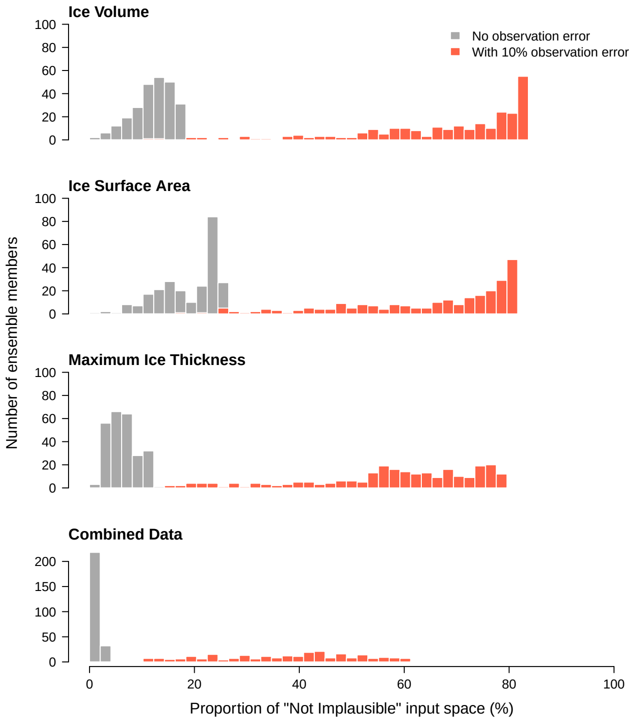

Fig. 8. Histograms of V, the estimated volume of "not implausible" ($I < 3$) input space, for each of the 250 members of the ensemble. The case with no observational error (grey bars) shows a much stronger potential constraint than when a representative observational uncertainty of 10 % of the maximum value of the ensemble is included (red bars).

inputs is degraded, with a much wider range of constraints, and even the combined data only able to constrain the volume of the plausible space to 60 % of its original size. In individual inputs, the ability to constrain is also greatly reduced, with some data unable to constrain inputs at all. Only maximum ice thickness now offers a genuine chance to significantly constrain a single input: FEF.

The extent to which an input parameter can be constrained when we observe an output depends upon (a) the output type, and (b) the location of the output (and hence input) within the ensemble. This is because the relationship between inputs and outputs changes across the ensemble. We measure the extent to which constraint of inputs is possible, in Fig. 9. Here, for each way of summarising the data, the range R of inputs found to be not implausible is plotted on the y axes, against each input on the x axes. We see that the true position of the ensemble member in input space has little systematic impact, except in a few cases. Most apparent is the effect of FEF parameter: it is easier to constrain FEF, PDFFI, and to some extent NSLR and PDDFS if the value of FEF is either high or low. If the true value of FEF is low, it is possible to constrain it to a very small region of input space.

4 Discussion

In a set of "perfect simulator/observation" experiments across an ensemble of the simulator, we find that the true location of the input within the input parameter space does impact on our ability to constrain the simulator given its output. If the entire ensemble input space is initially plausible, any one of the ensemble members might be a candidate for a future observation of the system. The location of the best input in input space will have a powerful impact on how well we might constrain the simulator. We therefore find some considerable uncertainty as to the potential constraint of the input parameters.

Any systematic observational or simulator discrepancy uncertainty will have the effect of shifting the estimate of best input $\boldsymbol{x}^*$ in input parameter space, compared to the "no uncertainty" case of our example. It is therefore crucial that observational or simulator discrepancy uncertainty is included in the assessment of potential observations, in order to avoid overestimation of the value of data. Our example shows how observational uncertainty can seriously degrade our ability to use observations to constrain the inputs of a simulator, and also highlights the importance of simulating observational error in any "observation system simulation" experiment. As the observational and discrepancy uncertainties are equivalent in the implausibility calculation (Eq. 3), any study of the value of observational data must also take into account the potential value of reducing the simulator discrepancy – in effect, of improving the simulator.

Finding information about a particular input from other sources might also give us useful information with which to constrain other inputs. For instance, in our example learning about one particular input, FEF, would offer a stronger constraint for other parameters. The flexible nature of the emulator allows us to simulate learning about any subset of parameters, and to visualise the impact of that information on the plausibility of the input space. This could be a powerful technique in the process of simulator development.

Our method is perhaps most useful when we are quite uncertain about a good set of inputs. This is because the metric is defined relative to our prior knowledge about what constitutes a good parameter space. The larger this prior parameter space, the easier it is for data to be useful in constraining it. It is important therefore that we carefully elicit prior distributions for the parameters, in order not to overinflate the relative utility of a data set in constraining a parameter. The prior distributions should accurately represent the prior uncertainty of the modeller.

Our method could be useful in informing the design of observational strategies in situations where observing the true system is expensive or time consuming. Our approach could be used to prioritise observations in order to maximally constrain simulators, or in order to model how collecting data of previously unobserved phenomena might benefit simulator development. As a standard analysis technique, our example

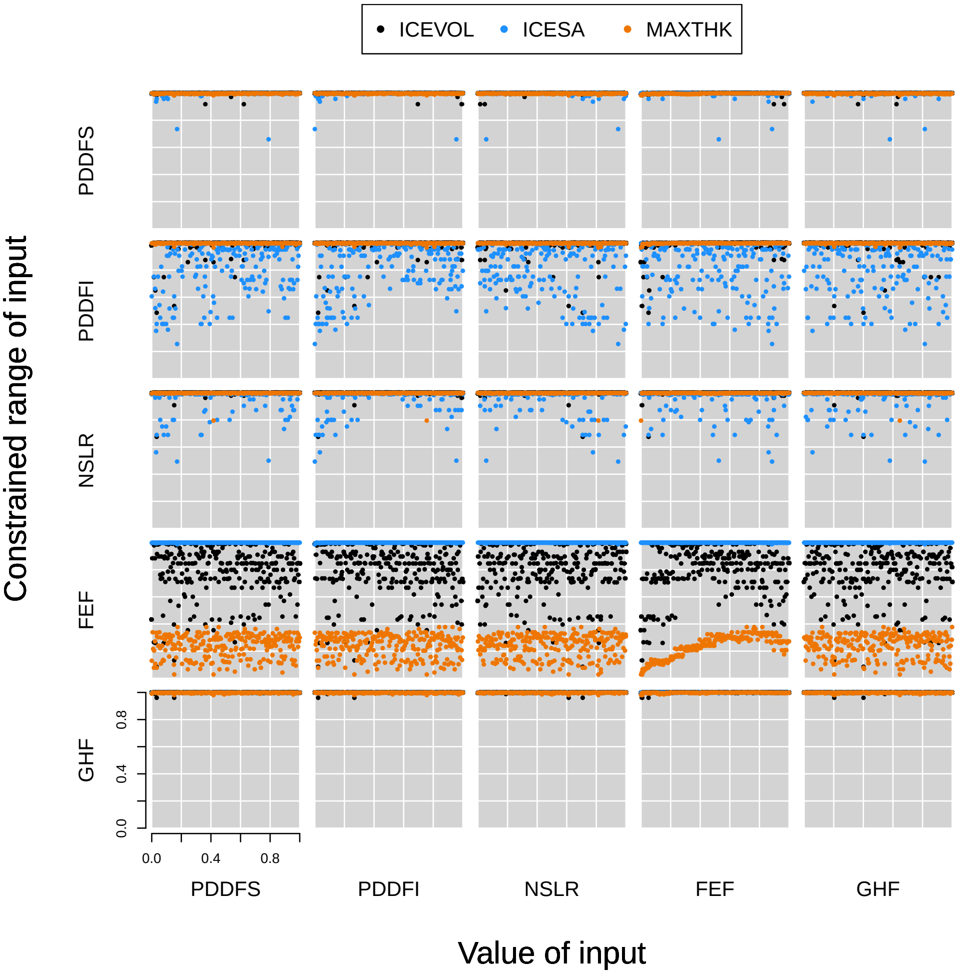

Fig. 9. The extent to which an input can be marginally constrained when we observe an output depends upon (a) the output type, and (b) the location of the output (and hence input) within the ensemble. Here we plot the extent to which an input can be constrained (height on the y axes), against its position in the input space of the ensemble (x axes), for the three different output summaries. We see that, for example, input parameter FEF can be very strongly constrained if the true value of FEF is low.

could be extended to show which observational data would maximally constrain future projections made by the simulator. In addition, the technique could be used to show how best to summarise data, in order to better constrain inputs. Our metric also offers a useful measure of sensitivity of simulator outputs to inputs, which may be useful in a simulator development process.

Care must be taken to ensure that the simulator discrepancy term realistically incorporates all plausible differences between the simulator and reality, and that the ensemble is drawn from a wide enough distribution in input space to accommodate any plausible combination of simulator output and model discrepancy. If the simulator discrepancy is poorly modelled (i.e. there was an “unknown unknown”), the estimated ability of data to constrain the simulator could be in error. It might rule out all of the input parameter space, for example if a real-world observation were to lie far from any simulated observation. A poorly specified discrepancy could also lead to less input space being ruled out than suggested by using the ensemble itself as pseudo-observations.

Any method of summarising a set of volumes (e.g. “not implausible” regions) in high dimensional space, will be inadequate when projected onto a two dimensional surface for visualisation on the printed page. We welcome further developments in visualisation techniques.

5 Conclusions

We have introduced a method for quantifying the upper bound of the potential of an observational data set to constrain the input of a computationally expensive simulator. Demonstrating the method on an ice sheet simulator, we find that we can identify a subset of simulator inputs which can be constrained by summaries of our observed field variable. The extent of that constraint varies between ensemble members. We find that if there were no observational or simulator discrepancy uncertainty and the true observations lay within that simulated by our model, we could rule out as implausible at least around 95 % of the input space covered by the initial ensemble. However, when a representative observational uncertainty (1 standard deviation) of around 10 % of the maximum ensemble value is assumed, we find that we are able to rule out at least around 40 % of the initial volume of input space.

We find that different ways of summarising our observational data might offer different, and potentially strong constraints for different input parameters. This means that a single observational data set might have more potential to calibrate a simulator than apparent at first glance. However, in general, the data (especially when observational error is considered), does not offer as strong a constraint on the marginal range of individual inputs as was expected by the authors before this experiment was run. This highlights the importance of a priori knowledge about the input parameters as an important constraint when using simulators to make predictions. There is some optimism however, that stronger constraints are possible when using multiple data sets for constraint, suggesting the importance of using multiple and varied data sets, with which to calibrate the simulator. This data might not correspond to the main output of interest, but nevertheless could contribute considerably to the constraint of the simulator.

Acknowledgements. This work was supported by funding from the ice2sea programme from the European Union 7th Framework Programme, grant number 226375. Ice2sea contribution number 154. We would like to thank referees M. Crucifix and I. Vernon for helping to improve the manuscript. We also thank those colleagues that made suggestions of relevant literature via the social networking website Twitter.

Edited by: D. Roche

References

Bamber, J., Layberry, R. L., and Gogineni, P.: A new ice thickness and bed data set for the Greenland ice sheet 1. Measurement, data reduction, and errors, J. Geophys. Res., 106, 33–773, 2001.

Craig, P., Goldstein, M., Seheult, A., and Smith, J.: Bayes linear strategies for history matching of hydrocarbon reservoirs, in: Bayesian Statistics 5, edited by: Bernardo, J., Berger, J., Dawid, A., and Smith, A., 69–95, Clarendon Press, Oxford, UK, 1996.

Craig, P., Goldstein, M., Seheult, A., and Smith, J.: Pressure matching for hydrocarbon reservoirs: a case study in the use of Bayes linear strategies for large computer experiments, in: Case studies in Bayesian statistics, edited by: Gatsonis, C., Hodges, J., Kass, R., McCulloch, R., Rossi, P., and Singpurwalla, N., vol. 3, 36–93, Springer-Verlag, New York, USA, 1997.

Craig, P., Goldstein, M., Rougier, J., and Seheult, A.: Bayesian forecasting for complex systems using computer simulators, J. Am. Stat. Assoc., 96, 717–729, 2001.

Hankin, R.: Introducing BACCO, an R package for Bayesian analysis of computer code output, J. Stat. Softw., 14, 1–21, 2005.

Kennedy, M. and O'Hagan, A.: Bayesian calibration of computer models, J. Roy. Stat. Soc.: Series B (Statistical Methodology), 63, 425–464, 2001.

McKay, M., Beckman, R., and Conover, W.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics, 21, 239–245, 1979.

Oakley, J.: Bayesian uncertainty analysis for complex computer codes, Ph.D. thesis, University of Sheffield, Sheffield, UK, 1999.

Oakley, J. and O'Hagan, A.: Probabilistic sensitivity analysis of complex models: a Bayesian approach, J. Roy. Stat. Soc.: Series B (Statistical Methodology), 66, 751–769, 2004.

Partridge, D. G., Vrugt, J. A., Tunved, P., Ekman, A. M. L., Struthers, H., and Sorooshian, A.: Inverse modelling of cloud-aerosol interactions – Part 2: Sensitivity tests on liquid phase clouds using a Markov chain Monte Carlo based simulation approach, Atmos. Chem. Phys., 12, 2823–2847, doi:10.5194/acp-12-2823-2012, 2012.

Payne, A.: A thermomechanical model of ice flow in West Antarctica, Clim. Dynam., 15, 115–125, 1999.

Pukelsheim, F.: The three sigma rule, American Statistician, 48, 88–91, 1994.

R Core Team: R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org, ISBN 3-900051-07-0, 2012.

Rougier, J.: Probabilistic inference for future climate using an ensemble of climate model evaluations, Climatic Change, 81, 247–264, 2007.

Rutt, I., Hagdorn, M., Hulton, N., and Payne, A.: The Glimmer community ice sheet model, J. Geophys. Res., 114, F02004, doi:10.1029/2008JF001015, 2009.

Saltelli, A., Chan, K. and Scott, E. (Eds.): Sensitivity analysis, vol. 134, Wiley New York, 2000.

Santner, T., Williams, B., and Notz, W.: The design and analysis of computer experiments, Springer, 2003.

Sexton, D. and Murphy, J.: Multivariate probabilistic projections using imperfect climate models. Part II: robustness of methodological choices and consequences for climate sensitivity, Clim. Dynam., 38, 1–16, 2543–2558, doi:10.1007/s0038201112098, 2012.

Sexton, D., Murphy, J., Collins, M., and Webb, M.: Multivariate probabilistic projections using imperfect climate models part I: outline of methodology, Clim. Dynam., 38, 2513–2542, doi:10.1007/s0038201112089, 2012.

Stone, E. J., Lunt, D. J., Rutt, I. C., and Hanna, E.: Investigating the sensitivity of numerical model simulations of the modern state of the Greenland ice-sheet and its future response to climate change, The Cryosphere, 4, 397–417, doi:10.5194/tc-4-397-2010, 2010.

Tebaldi, C. and Knutti, R.: The use of the multi-model ensemble in probabilistic climate projections, Philosophical Transactions of the Royal Society A: Mathematical, Phys. Eng. Sci., 365, 2053–2075, 2007.

Vernon, I., Goldstein, M., and Bower, R.: Galaxy formation: a Bayesian uncertainty analysis, Bayesian Analysis, 5, 619–669, 2010.

Wilkinson, R.: Bayesian calibration of expensive multivariate computer experiments, Large-Scale Inverse Problems and Quantification of Uncertainty, 195–215, 2011.

Ziehn, T., Scholze, M., and Knorr, W.: On the capability of Monte Carlo and adjoint inversion techniques to derive posterior parameter uncertainties in terrestrial ecosystem models, Global Biogeochem. Cy., 26, GB3025, doi:10.1029/2011GB004185, 2012.