Robust Experimental Designs for Model Calibration

Arvind Krishna

V. Roshan Joseph

H. Milton Stewart School of Industrial and Systems Engineering,
Georgia Institute of Technology, Atlanta, GA 30332

Shan Ba

LinkedIn Corporation, Sunnyvale, CA 94085

William A. Brenneman

William R. Myers

Procter & Gamble Company, Cincinnati, OH 45040

Abstract

A computer model can be used for predicting an output only after specifying the values of some unknown physical constants known as calibration parameters. The unknown calibration parameters can be estimated from real data by conducting physical experiments. This paper presents an approach to optimally design such a physical experiment. The problem of optimally designing a physical experiment, using a computer model, is similar to the problem of finding an optimal design for fitting nonlinear models. However, the problem is more challenging than the existing work on nonlinear optimal design because of the possibility of model discrepancy, that is, the computer model may not be an accurate representation of the true underlying model. Therefore, we propose an optimal design approach that is robust to potential model discrepancies. We show that our designs are better than the commonly used physical experimental designs that do not make use of the information contained in the computer model and other nonlinear optimal designs that ignore potential model discrepancies. We illustrate our approach using a toy example and a real example from industry.

Keywords: Bayesian calibration, Computer experiments, Physical experiments, Space-filling design, Uncertainty quantification.

1 Introduction

A physical system can be explored or optimized by conducting experiments, but they can be expensive and time consuming. The whole field of experimental design in statistics is focused on how to perform these experiments in an efficient way so that maximum information about the system can be obtained with minimum cost (Wu & Hamada 2011). Another way to reduce the experimental cost is to develop mathematical models that can mimic the physical system and explore the system through simulations (Santner et al. 2018). These mathematical models can be very complex, such as a system of partial differential equations, which needs to be solved numerically. Computer implementation of the mathematical model using a numerical solver is sometimes called a computer model. Exploration using computer models is useful and can provide conclusive results only if they are good in representing the physical system. However, the mathematical model and thus, the computer model is only an approximation to the complex phenomenon that we are trying to explore in the physical system. Therefore, the computer simulations should only be used to assist the physical experimentation and physical experiments should always be performed to validate the computer models. This article examines how to perform a physical experiment when a computer model is available to the investigator.

A computer model can contain unknown parameters. Choosing them based on the physical experimental data can make the computer models closer to reality. This approach is known as model calibration and the unknown parameters are often referred to as calibration parameters (Box & Hunter 1962). Therefore, one possible approach to physical experiments is to design them in such a way that the calibration parameters can be estimated efficiently from data. Since the computer models are often nonlinear in the calibration parameters, one can use results from the nonlinear optimal design theory to design such experiments (Silvey 1980). One of the major pitfalls of this approach is that the optimal design is overly dependent on the computer model and is not robust against possible model violations. However, detecting such possible violations is at the core of the model calibration problem and is our main aim. Thus, we need to design experiments that account for the computer model but at the same time are robust to the misspecifications of the model.

The importance of designing experiments robust to the model assumptions has long been recognized in the literature since Box & Draper (1959). A good account of the follow-up research

on their seminal paper and other related developments can be found in the review by Chang & Notz (1996). Unfortunately, most of these approaches rely on an alternative class of possible models, the specification of which is non-trivial. Therefore, the research in this area has not led to practically implementable solutions other than in very simple settings such as adding a center point when fitting a plane, etc. Moreover, the literature seems to be scarce on model-robust designs for nonlinear models. Nonlinear models add complexity to the problem because the optimal designs become functions of the unknown parameters. Robust designs can be developed by expressing the uncertainties in the unknown parameters through a prior distribution and using Bayesian optimal designs (Chaloner & Verdinelli 1995) or robust-Bayesian optimal designs (Dror & Steinberg 2006). Introducing model uncertainty into this framework makes the problem even harder to solve. Furthermore, most of the existing work focuses on simple nonlinear models such as a logistic regression model and not on the complex computer models that we are interested in. Computationally expensive computer models add another layer of complexity because they are known only in the places where the computer simulations have been performed. Therefore, we also need to entertain uncertainties arising due to incomplete knowledge about the true computer model output.

We are not the first to look into the problem of designing physical experiments when computer models are available. Based on the Bayesian model calibration framework of Kennedy & O'Hagan (2001), optimal designs for both computer and physical experiments using integrated mean squared prediction error have been proposed by Leatherman et al. (2017). However, it has been recognized that the Kennedy and O'Hagan model has severe identifiability issues (Tuo & Wu 2015, Plumlee 2017) and so, optimal designs based on their model can inherit similar problems. Arendt et al. (2016) have proposed designing physical experiments to mitigate the identifiability issues in the Kennedy and O'Hagan model, but their procedure is computationally intensive. In this article, we will propose a much simpler approach to deal with the identifiability issue. Ranjan et al. (2011) and Williams et al. (2011) have proposed follow-up experimental designs for model calibration, but not an initial design that we plan to develop here.

This article is organized as follows. In Section 2, we discuss the methodology of obtaining a robust experimental design for model calibration. In Section 3, we perform simulations on a toy example to illustrate the robustness of our proposed design to model discrepancy. In Section 4, we apply our design methodology to a problem relating to the diaper line from the Procter & Gamble

(P&G) company. We conclude the article with some remarks in Section 5.

2 Robust Experimental Designs

We will first explain how to develop experimental designs that are robust to model-form uncertainties. Then we will explain how to make them robust to parameter uncertainties. Finally, we will explain how to incorporate computer model approximation uncertainties into this framework. These are now discussed in the following three subsections.

2.1 Model-form uncertainties

Let y be the output of the physical system and $\boldsymbol{x} = \{x_1, ..., x_p\}$ the set of inputs. Following Kennedy & O'Hagan (2001), we model the output as

$$y = f(\boldsymbol{x}; \boldsymbol{\eta}) + \delta(\boldsymbol{x}) + \epsilon, \tag{1}$$

where $f(\cdot;\cdot)$ is the computer model, $\boldsymbol{\eta} = \{\eta_1, ..., \eta_q\}$ the set of unknown calibration parameters, $\delta(\boldsymbol{x})$ the discrepancy function, and $\epsilon \overset{i.i.d}{\sim} \mathcal{N}(0, \sigma^2)$ the random error. For the moment, we will assume that the computer model is an easy-to-evaluate function or that the complex computer model has been replaced by an easy-to-evaluate surrogate model whose uncertainties can be ignored. Later we will see how such uncertainties can also be included in our approach.

As mentioned earlier, the model in (1) has identifiability issues in the sense that for any value of $\boldsymbol{\eta}$ we can find a $\delta(\boldsymbol{x})$ to get the same prediction (Tuo & Wu 2015, Plumlee 2017) and thus, $\boldsymbol{\eta}$ and $\delta(\boldsymbol{x})$ cannot be estimated based on the data on y alone unless some additional assumptions are imposed. Therefore, we will not directly use this model for developing the robust designs. We will put some belief in the computer model and hope that, if properly calibrated, we will not need the discrepancy term. Thus, we will first find an optimal design for estimating $\boldsymbol{\eta}$ ignoring the model discrepancy term. We will then separately find an optimal design for estimating $\delta(\boldsymbol{x})$ and then integrate them together. This approach follows the sequential model building strategy proposed by Joseph & Melkote (2009), where $\boldsymbol{\eta}$ is estimated by first ignoring the discrepancy. The discrepancy term is added only if it is necessary and if added, it is estimated by fixing $\boldsymbol{\eta}$ at its initial estimate. This mitigates the identifiability issues that are present in a joint estimation procedure.

Thus, first consider the case with no discrepancy, that is, $\delta(\boldsymbol{x}) = 0$. Our aim is to choose a design to efficiently estimate $\boldsymbol{\eta}$ from the model

$$y = f(\boldsymbol{x}; \boldsymbol{\eta}) + \epsilon. \tag{2}$$

Suppose we have a total budget of N runs. We will use n runs to efficiently estimate $\boldsymbol{\eta}$ and the remaining $N - n$ runs to estimate the discrepancy function. Choice of n depends on how much confidence we have in the computer model form. If we are fully confident that there is no discrepancy, then $n = N$. On the other hand, if we have no confidence in the computer model, then $n = 0$, that is, we will not use the computer model to design the physical experiment. Let $\gamma \in [0, 1]$ be a parameter that measures the experimenter's confidence in the computer model form. Then, $n = [\gamma N]$, the nearest integer to γN. In the absence of any knowledge about the confidence in the computer model, we recommend choosing $\gamma = 2q/N$. This recommendation is based on the fact that at least q runs are needed to estimate q parameters in the nonlinear model (e.g. Masoudi et al. 2019). Thus using twice the minimum number of runs, we are likely to have two replicates for each run, which can be used for estimating the unknown error variance, σ^2.

The *approximate* optimal design for estimating $\boldsymbol{\eta}$ can be viewed as a discrete probability distribution in the experimental region $\mathcal{X}$ at points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ with probabilities $\{w_1, \ldots, w_n\}$ (Kiefer 1985). Denote the approximate optimal design (or the distribution) by ξ, which contains the design points as well as their probabilities. The Fisher Information matrix of $\boldsymbol{\eta}$ is given by:

$$M(\xi; \boldsymbol{\eta}) = \sum_{i=1}^{n} w_i \nabla f(\boldsymbol{x}_i; \boldsymbol{\eta}) \nabla f(\boldsymbol{x}_i; \boldsymbol{\eta})^T, \tag{3}$$

where $\nabla f(\boldsymbol{x}_i; \boldsymbol{\eta})^T = \left(\frac{\partial f(\boldsymbol{x}_1; \boldsymbol{\eta})}{\partial \eta_1}, \ldots, \frac{\partial f(\boldsymbol{x}_1; \boldsymbol{\eta})}{\partial \eta_q} \right)$. Now an approximate locally D-optimal design can be obtained by maximizing the determinant of $M(\boldsymbol{\eta})$ for a given value of $\boldsymbol{\eta}$. Suppose $\boldsymbol{\eta}_0$ is a guess value of $\boldsymbol{\eta}$. Then, we can obtain the approximate design as

$$\xi^* = \arg\max_{\xi} |M(\xi; \boldsymbol{\eta}_0)|. \tag{4}$$

In general, this is a very difficult optimization problem. Here we will use the metaheuristic algorithm proposed in Masoudi et al. (2019), which is implemented in the R package `ICAOD` (Masoudi

et al. 2020).

The approximate design can be converted to an exact design by rounding $nw_1, \ldots, nw_n$ to the nearest integers. However, the total number of design points after rounding need not be equal to n unless special care is taken (Pukelsheim & Rieder 1992). Instead of the rounding approach, here we propose a resampling-based approach, that is, sample with replacement from $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ with probabilities $\{w_1, \ldots, w_n\}$. However, the commonly used random resampling methods are not suitable for generating an optimal design because the results can vary due to the randomness involved in sampling. Huang et al. (2020) recently proposed a deterministic resampling method known as Importance Support Points (ISP), which finds an optimal set of samples with equal weights to approximate a discrete probability distribution. The optimal samples are obtained as the solution to the following optimization problem:

$$\{\tilde{\boldsymbol{x}}_i\}_{i=1}^n \in \underset{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_n \in \{\boldsymbol{x}_j\}_{j=1}^n}{\arg\min} \frac{2}{n}\sum_{i=1}^n\sum_{j=1}^n w_j\|\boldsymbol{u}_i - \boldsymbol{x}_j\|_2 - \frac{1}{n^2}\sum_{i=1}^n\sum_{j=1}^n\|\boldsymbol{u}_i - \boldsymbol{u}_j\|_2 \,. \tag{5}$$

Thus, we obtain the final design $\mathcal{D}_\eta = \{\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_n\}$ for efficiently estimating $\boldsymbol{\eta}$ by doing an ISP resample on ξ^* in which some of the points may be replicated.

Now that we have an optimal design to estimate the calibration parameters, we can focus our attention on the design to estimate the potential model discrepancy, $\delta(\boldsymbol{x})$. Let us denote the design by $\mathcal{D}_\delta$, which contains $N - n$ runs. So the final design will be $\mathcal{D} = \mathcal{D}_\eta \bigcup \mathcal{D}_\delta$.

Kennedy & O'Hagan (2001) proposed to nonparametrically estimate $\delta(\boldsymbol{x})$ using a Gaussian process. However, a Gaussian process model contains a set of unknown correlation parameters. These correlation parameters are nonlinear, which brings up the same issue as the calibration parameters. As before, we can guess the values of the correlation parameters and try to develop locally optimal designs. However, there is something peculiar about these correlation parameters. The optimal design criteria such as the integrated mean squared error (Leatherman et al. 2017) are dominated by settings that produce low values of correlations (Joseph, Gu, Ba & Myers 2019). Thus, we only need to focus on a setting of the correlation parameters that minimizes the correlation. Johnson et al. (1990) have shown that in such limiting cases the D-optimal designs will reduce to maximin designs and G-optimal designs to minimax designs. In fact, these space-filling designs can be independently motivated using geometric considerations and are known to be robust to modeling choices. Since our objective is to develop model-robust designs, they seem to be

a perfect fit for our problem. Besides maximin and minimax, there are many choices for space-filling designs (Joseph 2016). In this article, we will illustrate the methodology using maximum projection (MaxPro) designs (Joseph et al. 2015), but we will keep this choice flexible for the experimenter. It is important to note that these space-filling designs should be generated to optimally *augment* the existing points in $\mathcal{D}_\eta$ and should not be generated independent of $\mathcal{D}_\eta$.

The proposed method for generating the design is summarized in Algorithm 1.

Algorithm 1 : Generating the physical experimental design, accounting for model-form uncertainties.

1: Input $f(\boldsymbol{x}, \boldsymbol{\eta}), N, \gamma, \boldsymbol{\eta}_0$
2: $n \leftarrow [\gamma N]$
3: Find ξ^*, as in (4) {`R` package: `ICAOD` }
4: Find $\mathcal{D}_\eta$, as in (5), using ISP resampling on ξ^*
5: Find $\mathcal{D}_\delta$, a space-filling design with $N - n$ runs, to augment $\mathcal{D}_\eta$ {`R` package: `MaxPro` }
6: Output $\mathcal{D}_\eta \bigcup \mathcal{D}_\delta$

As a simple example, suppose the computer model is a linear model given by $f(\boldsymbol{x}; \boldsymbol{\eta}) = \eta_0 + \eta_1 x_1 + \eta_2 x_2 + \eta_3 x_1 x_2$. D-optimal design for estimating this model is a 2^2 full factorial design. Suppose we have a total budget for $N = 13$ runs, and there is no knowledge of the confidence in the computer model. If we choose $\gamma = 2q/N = 8/N$, then $n = [\gamma N] = 8$ and we will have two replicates for each of the four points in the 2^2 full factorial design. The remaining $N - n = 5$ runs can be chosen to augment the eight runs using a space-filling criterion. For example, if we use the MaxPro criterion, then the augmented design can be obtained sequentially by adding one point at a time using the `MaxProAugment` function in the R package `MaxPro` (Ba & Joseph 2018).

Figure 1 (right) shows the 13-run design obtained using the Maxpro design for augmentation along with maximin (left) and minimax (center) augmentation strategies. We can see that all these designs will provide protection against possible departures from the linear model assumption. At first glance, the maximin design seems most suitable for a physical experiment as it has fewer factor levels. However, in a high-dimensional space, this design may not give a good validation set as the points will occupy mostly the corners and boundaries of the hypercube. On the other hand, a MaxPro design simultaneously ensures space-fillingness in the full dimensional space as well as in all lower dimensional subspaces. Moreover, it can easily be extended to incorporate qualitative factors (Joseph, Gul & Ba 2019). However, a disadvantage of the MaxPro design is that it can produce too many levels for the factors, which can be inconvenient for a physical experiment. One

quick remedy to this problem is to treat the factors as discrete-numeric with a specified number of levels in the `MaxProAugment` function (Ba & Joseph 2018).

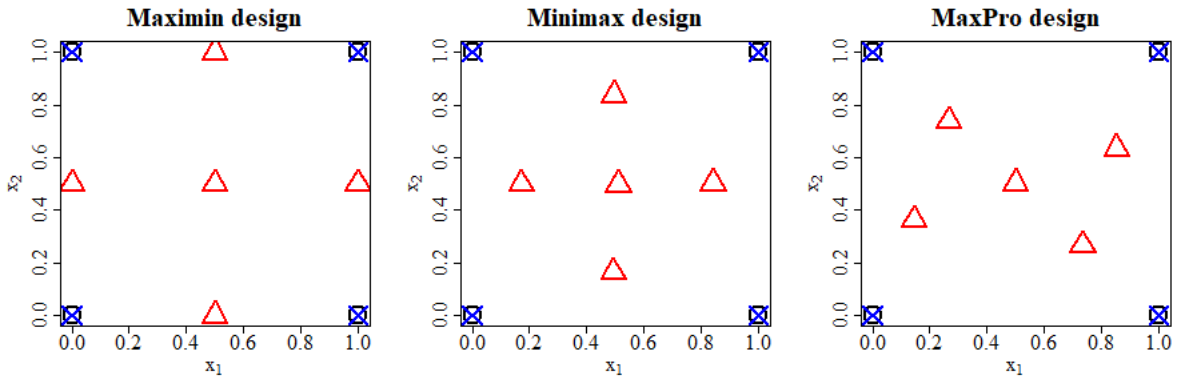

Figure 1: Three model-robust designs in 13 runs for estimating a linear model in two variables. The replicates of the optimal design points are shown as circles and crosses, and the space-filling points as triangles.

Now consider a nonlinear model:

$$f(\boldsymbol{x};\eta) = \exp\{-\eta(x_1 - 1.5x_2)^2\} + \exp\{-2\eta(x_1 + x_2 - 0.7)^2\}, \tag{6}$$

where $x_1, x_2 \in [0,1]^2$. Suppose that we have a budget to perform eight physical experiments. Assume $\eta_0 = 0.5$, and that there is no information about the confidence in the computer model. Then, we choose $\gamma = 2q/N = 2/N$. This leads to $n = [\gamma N] = 2$. Then, the exact locally D-optimal optimal design is a one-point design at $\{(0.50, 1.00)\}$ with two replicates. The remaining six runs are obtained by augmenting these two points with the MaxPro design. The final model-robust optimal design is shown in Figure 2 (left). For comparison, we have also shown a 2^2 design with two replicates in Figure 2 (right), which would be a reasonable choice to make when we don't have a computer model. Furthermore, we also show the D-optimal design with eight replicates in Figure 2 (center).We call this design as the "pure computer model" design.

If there is no model discrepancy, the "pure computer model" design is likely to provide the most accurate estimate of η. On the other hand, the full factorial design is likely to give the least accurate estimate of η. In the presence of model discrepancy, the "pure computer model" design is likely to perform poorly as it does not contain any space-filling points for estimating model discrepancy. However, in both cases - presence or absence of model discrepancy - our proposed

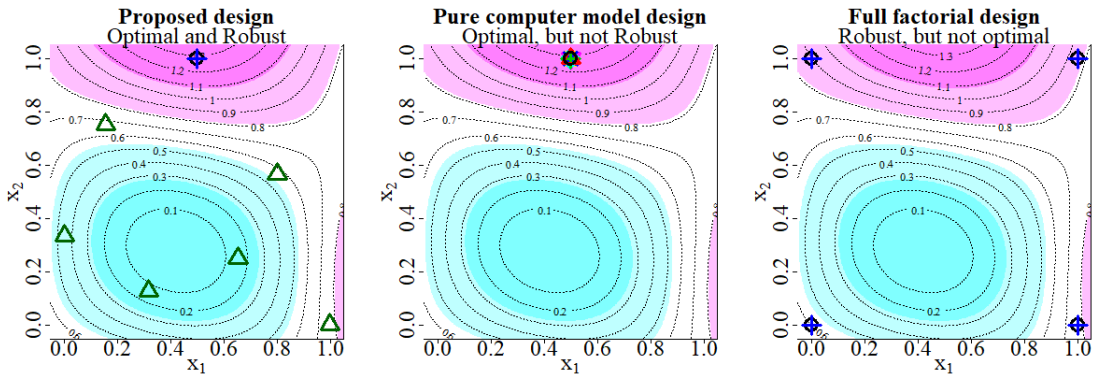

Figure 2: Comparison of our proposed design with the "pure computer model" design, and full factorial design, over the computer model gradient contour. The replicates of the D-optimal design points are shown as circles and crosses and the space-filling points as triangles.

design, though it may not be the best, is likely to provide a relatively accurate calibrated model. This is because it contains both - the D-optimal design points optimal for estimating η and the space-filling design points optimal for estimating the model discrepancy.

2.2 Parameter uncertainties

A weakness of the locally optimal designs is that the solution depends on the guessed value of $\boldsymbol{\eta}$. If the guessed value is not close to the (unknown) true value, the results may not be accurate especially when the model is highly nonlinear. As discussed in the introduction, a natural way to address these uncertainties is to use a Bayesian approach by placing a prior on $\boldsymbol{\eta}$. So let $p(\boldsymbol{\eta})$ be the prior distribution. The $\boldsymbol{\eta}_0$ that we used in the previous section could be viewed as the mean or mode of this prior distribution.

Following Dror & Steinberg (2006), now we will generate multiple values for $\boldsymbol{\eta}$:

$$\boldsymbol{\eta}_i \sim p(\boldsymbol{\eta}), \quad i = 1, \ldots, m.$$

We can find approximate locally D-optimal designs for each of these values as described in the previous section to obtain $\xi_1, \ldots, \xi_m$. Thus, we have a total of nm points with nm probabilities. After re-scaling all the weights to sum to one, we can again use the ISP resampling method to obtain the desired n-point optimal design $\mathcal{D}_\eta$.

As we need to find ξ^i for $i = 1, \ldots, m$, the procedure is much more computationally expensive

than before. One idea to reduce the computational burden is to sample $\boldsymbol{\eta}_i$ using support points (Mak & Joseph 2018) instead of random values from $p(\boldsymbol{\eta})$, because support points will be able to represent the prior distribution with fewer points than a Monte Carlo sample. Thus we can use a much smaller m. The procedure is summarized in Algorithm 2.

Algorithm 2 : Generating the physical experimental design, accounting for model-form and parameter uncertainties.

Input $f(\boldsymbol{x}, \boldsymbol{\eta}), N, \gamma, p(\boldsymbol{\eta})$
$n \leftarrow [\gamma N]$
Find m realizations of $\boldsymbol{\eta}$ from $p(\boldsymbol{\eta})$ {R package: `support`}
for $i \in \{1, \cdots, m\}$ **do**
    $\boldsymbol{\eta}_0 \leftarrow \boldsymbol{\eta}_i$
    Find ξ_i^*, as in (4) {R package: `ICAOD` }
    $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i / m$
end for
Find $\mathcal{D}_\eta$, as in (5), using ISP resampling on ξ^*
Find $\mathcal{D}_\delta$, a space-filling design with $N - n$ runs, to augment $\mathcal{D}_\eta$ {R package: `MaxPro` }
Output $\mathcal{D}_\eta \bigcup \mathcal{D}_\delta$

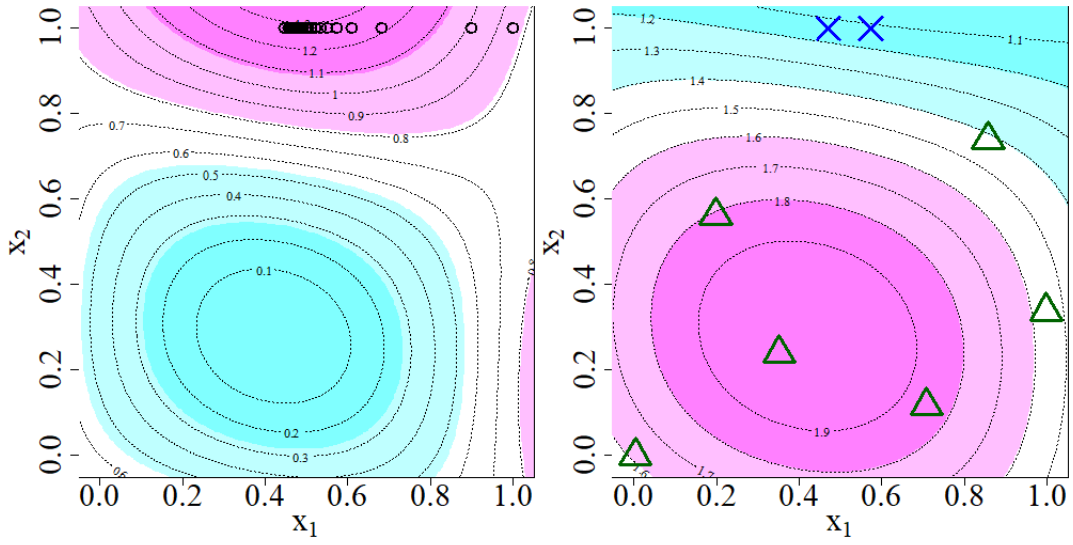

Figure 3: (left): Approximate locally D-optimal design points incorporating parameter uncertainties shown over the gradient contour of the computer model for $\eta = 0.5$; (right): The desired $N = 8$-run design over the computer model contour for $\eta = 0.5$, the crosses correspond to the exact D-optimal design for estimating $\boldsymbol{\eta}$, and the triangles correspond to the space-filling design.

Consider again the toy example with one calibration parameter used in the previous section. We assume that the calibration parameter has a prior distribution $\eta \sim \mathcal{N}(0.5,\ 0.2^2)$. We obtain $m = 20$

support points to represent this distribution using the R package `support` (Mak 2019). For each of the $m = 20$ realizations, we obtain two-point approximate locally D-optimal design $\xi_i^*, i \in \{1, \cdots, m\}$. The points corresponding to the $m = 20$ approximate optimal designs obtained are shown in Figure 3 (left).

Now we use ISP resampling to obtain $n = 2$ runs as shown by crosses in Figure 3 (right). This two-run design is augmented with the MaxPro design, shown by triangles in Figure 3 (right), to obtain the desired 8-run design. We observe that the design obtained is somewhat similar to the one obtained without incorporating parameter uncertainties in Figure 2 (left). However, unlike the design in Figure 2 (left), the two points of the D-optimal design for estimating η are a bit farther apart (instead of overlapping) to account for parameter uncertainty.

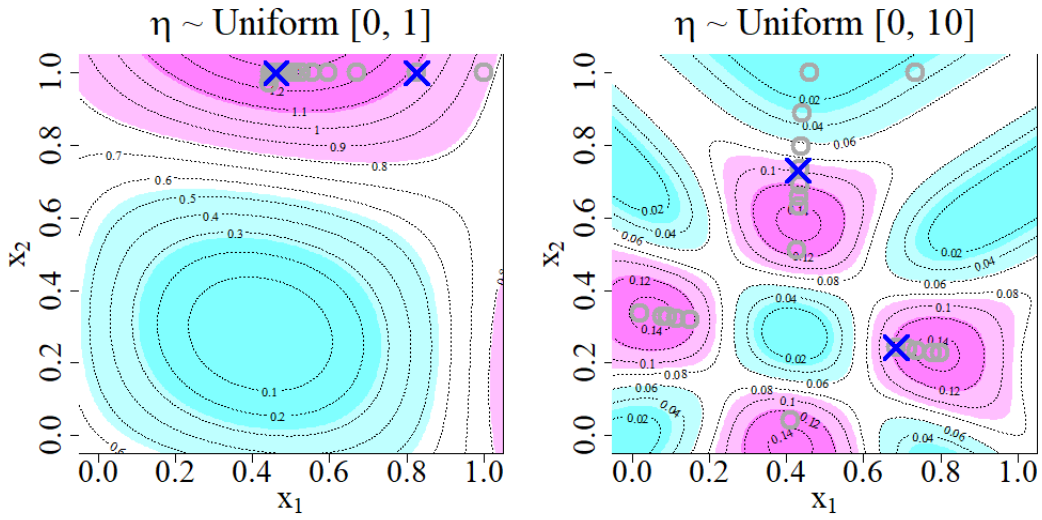

Figure 4: Approximate locally D-optimal design points (circles), and $\mathcal{D}_\eta$ (crosses) for (left): $\eta \sim \mathcal{U}[0, 1]$, over the computer model gradient contour for $\eta = 0.5$; (right): $\eta \sim \mathcal{U}[0, 10]$, over the computer model gradient contour for $\eta = 5$.

Now consider the case where the prior is misspecified. The left panel of Figure 4 shows the set of locally D-optimal designs (circles) and $\mathcal{D}_\eta$ selected using ISP resampling when the prior is $\mathcal{U}[0, 1]$. The results are quite similar to the design obtained earlier using the prior $\mathcal{N}(0.5,\, 0.2^2)$ except that the two selected design points are slightly more spaced-out. Now consider the prior $\eta \sim \mathcal{U}[0, 10]$, where the center of the distribution is far away from the previous center. The results are shown in the right panel of Figure 4. We can see that the $\mathcal{D}_\eta$ in Figure 3 is not useful anymore for efficiently estimating η. However, because of the space-filling design points, the overall design

is still good and would perform much better than a pure computer model design.

2.3 Surrogate model uncertainties

So far we had assumed that $f(\boldsymbol{x};\boldsymbol{\eta})$ is an easy-to-evaluate model. In reality the computer model can be very expensive to evaluate. In such cases, we will first perform a computer experiment to obtain an approximation of $f(\boldsymbol{x};\boldsymbol{\eta})$. The approximate model is called a surrogate model or an emulator. Although there exist many different methods to obtain the surrogate model, Gaussian process modeling (or kriging) seems to be the most popular choice because of its ability to provide uncertainty estimates (Sacks et al. 1989).

Let $\mathcal{S}$ denote the computer experimental design and $\boldsymbol{y}$ be the output values. Note that unlike in the physical experiment, the calibration parameters can be varied in the computer experiment. Thus, $\mathcal{S}$ has $p+q$ columns. For simplifying the notations, let $\boldsymbol{u} = (\boldsymbol{x},\boldsymbol{\eta})$ be the inputs in the computer experiment. Assume that $f(\cdot)$ is a realization of a Gaussian process:

$$f(\boldsymbol{u})|\boldsymbol{\eta} \sim GP(\mu, C(\boldsymbol{u};\cdot)),$$

where μ is the mean and $C(\boldsymbol{u};\boldsymbol{v}) = Cov\{f(\boldsymbol{u}), f(\boldsymbol{v})\}$ is the covariance function. See Santner et al. (2018) for details on Gaussian process modeling. Given the data, the posterior distribution of $f(\boldsymbol{u})$ is also a Gaussian process given by

$$f(\boldsymbol{u})|\boldsymbol{\eta},\boldsymbol{y} \sim GP(\hat{f}(\boldsymbol{u}), C(\boldsymbol{u};\cdot) - C(\boldsymbol{u};\boldsymbol{S})C^{-1}(\boldsymbol{S};\boldsymbol{S})C(\boldsymbol{S};\cdot)), \tag{7}$$

where $\hat{f}(\boldsymbol{u}) = \mu + C(\boldsymbol{u};\boldsymbol{S})C^{-1}(\boldsymbol{S};\boldsymbol{S})(\boldsymbol{y}-\mu\mathbf{1})$ is the surrogate model, $C(\boldsymbol{u};\boldsymbol{S})$ is the covariance vector with ith element $C(\boldsymbol{u};\boldsymbol{S}_i)$, $C(\boldsymbol{S};\boldsymbol{S})$ is the covariance matrix, and $\mathbf{1}$ is a vector of 1's.

Incorporating the surrogate model uncertainties into our design construction is conceptually very simple. We generate m samples from $p(\boldsymbol{\eta})$ and for each sample, we generate a realization of the function from (7):

$$\begin{aligned} \boldsymbol{\eta}_i &\sim p(\boldsymbol{\eta}), \\ f_i(\boldsymbol{u})|\boldsymbol{\eta}_i,\boldsymbol{y} &\sim p(f(\boldsymbol{u})|\boldsymbol{\eta}_i,\boldsymbol{y}) \end{aligned}$$

for $i = 1, \ldots, m$. Now we can proceed to find the optimal design in the same way as in the previous section except for one difference. The gradients needed for the sensitivity matrix need to be calculated numerically for each new realization of $f(\cdot)$. This makes the procedure computationally very expensive. The procedure is summarized in Algorithm 3.

Algorithm 3 : Generating the physical experimental design, accounting for model-form, parameter, and surrogate-model uncertainties.

1: Input $p(f(\boldsymbol{x}, \boldsymbol{\eta})|\boldsymbol{\eta}, \boldsymbol{y}), N, \gamma, p(\boldsymbol{\eta})$
2: $n \leftarrow [\gamma N]$
3: Find m realizations of $\boldsymbol{\eta}$ from $p(\boldsymbol{\eta})$ {R package: `support`}
4: **for** $i \in \{1, \cdots, m\}$ **do**
5: $\quad \boldsymbol{\eta}_0 \leftarrow \boldsymbol{\eta}_i$
6: $\quad$ Find a realization of $f(\cdot)$ from $p(f(\boldsymbol{x}, \boldsymbol{\eta})|\boldsymbol{\eta} = \boldsymbol{\eta}_0, \boldsymbol{y})$
7: $\quad$ Find ξ_i^*, as in (4) {R package: `ICAOD` }
8: $\quad \boldsymbol{w}_i \leftarrow \boldsymbol{w}_i/m$
9: **end for**
10: Find $\mathcal{D}_\eta$, as in (5), using ISP resampling on ξ^*
11: Find $\mathcal{D}_\delta$, a space-filling design with $N - n$ runs, to augment $\mathcal{D}_\eta$ {R package: `MaxPro` }
12: Output $\mathcal{D}_\eta \bigcup \mathcal{D}_\delta$

It is a good idea to make the physical experimental design a subset of the computer experiment design, that is, a nested design (Qian 2009). This avoids the confounding between the surrogate model approximation error and the discrepancy. To get a nested design, we have two options: (i) go back and run the computer simulations at the optimal physical experimental design points or (ii) choose the nearest points in $\mathcal{S}$ as the physical experimental design. If we use option (ii), there will be some loss of efficiency. Therefore, option (i) is preferred if the optimal design points are far away from $\mathcal{S}$ and it is feasible to run the computer simulation again. It is possible that the new simulations can change the surrogate model and hence the optimal design. So it seems some iterations will be needed to finalize the physical experimental design. However, we do not expect this to happen in most realistic cases unless there is too much uncertainty in the surrogate model approximation.

Consider again the toy example. For illustrative purposes, we assume that the computer model is too expensive to compute. We consider a 30-run MaxPro design in x_1, x_2, and η to develop a surrogate Gaussian process model. Instead of the computer model $f(\boldsymbol{x}; \boldsymbol{\eta})$, we use random realizations from the surrogate model $f_i(\boldsymbol{u})|\boldsymbol{\eta}_i, \boldsymbol{y}$, for $i = 1, \ldots, m$, to obtain the $m = 20$ approximate locally D-optimal designs (Figure 5 (left)). It can be seen that due to uncertainty in the surrogate

model prediction, the design points are more dispersed than in the case of no surrogate model uncertainty (Figure 3 (left)).

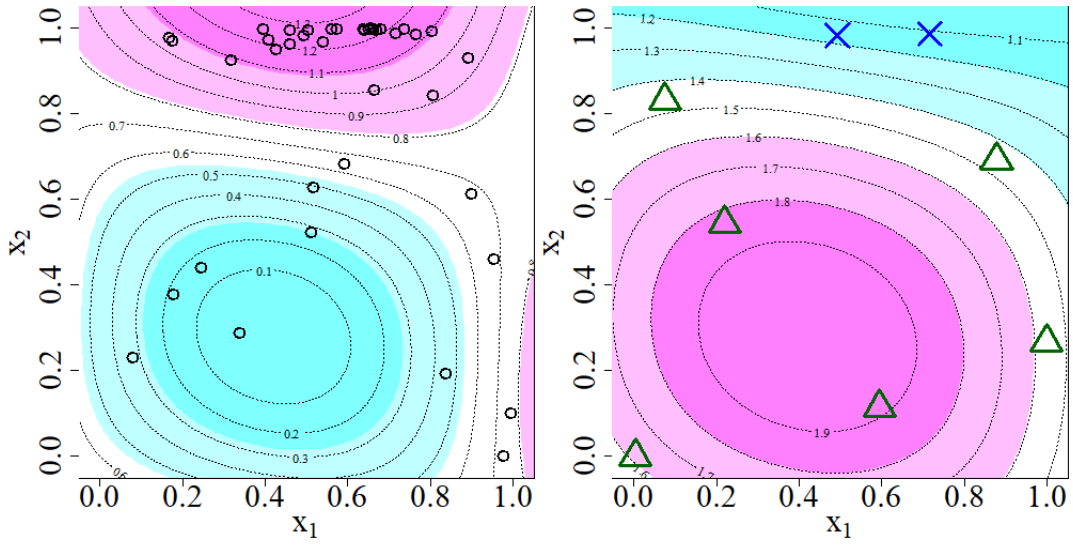

Figure 5: (left): Approximate locally D-optimal design points incorporating parameter and surrogate model uncertainties shown over the gradient contour of the computer model for $\eta = 0.5$; (right): The desired $N = 8$-run design over the computer model contour for $\eta = 0.5$, the crosses correspond to the D-optimal design for estimating $\boldsymbol{\eta}$, and the triangles correspond to the space-filling design.

As in the previous section, we use ISP resampling to obtain $n = 2$ runs as shown by crosses in Figure 5 (right). This two-run design is augmented with the Maxpro design, shown by triangles in Figure 5 (right), to obtain the desired 8-run design. It can be seen that with the addition of surrogate model uncertainty, the two-points of the D-optimal design for estimating $\boldsymbol{\eta}$ are farther away as compared to the case in the previous section (Figure 3 (right)).

3 Simulations

In this section we will investigate the robustness of the proposed design to potential model discrepancies. Consider the toy example in (6) again. Let the output be

$$y = f(\boldsymbol{x}; \eta) + \delta(\boldsymbol{x}) + \epsilon, \tag{8}$$

where $\epsilon \overset{i.i.d}{\sim} \mathcal{N}(0, 0.05^2)$. Assume that $\delta(\boldsymbol{x}) \sim GP(0, \tau^2 R(\cdot))$, where $R(\cdot)$ is a stationary correlation function and τ^2 the variance. We will use the Gaussian correlation function given by $R(\boldsymbol{h}) = \exp(-\theta\|\boldsymbol{h}\|^2)$ with $\theta = 10$. Now we will generate the outputs by increasing the magnitude of the model discrepancy (i.e., by increasing τ^2) and study the performance of the three designs shown in Figure 2.

Given the design and the output, estimation of the model in (8) is done in two steps. First, ignoring $\delta(\boldsymbol{x})$, the posterior distribution of η is found using Markov chain Monte Carlo (MCMC) simulations. Then, using the posterior samples η^i, $i = 1, \ldots, N$, the physical experiment output at a point $\boldsymbol{x}$ is estimated as:

$$\hat{f}(\boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} f(\boldsymbol{x}; \eta^i). \tag{9}$$

The discrepancy at each of the design points can be obtained as $\delta_i = y(\boldsymbol{x}_i) - \hat{f}(\boldsymbol{x}_i)$ for $i = 1, \ldots, n$. Let $\hat{\delta}(\boldsymbol{x})$ be the posterior mean of $\delta(\boldsymbol{x})$ given $\mathcal{D}$ and $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_n)'$. Then the bias-corrected calibrated model at a point $\boldsymbol{x}$ is given by:

$$\hat{y}(\boldsymbol{x}) = \hat{f}(\boldsymbol{x}) + \hat{\delta}(\boldsymbol{x}). \tag{10}$$

The prediction accuracy of the model is evaluated on a 500-point Sobol test dataset generated using the R package `randtoolbox` (Christophe & Petr 2019).

Figure 6 plots the root mean squared prediction error (RMSPE) with $\tau^2 = 0.0, 0.01, \ldots, 0.5$. We observe that, when there is no discrepancy, the "pure computer model" design is the best, as expected, followed closely by our proposed design. However, as the model discrepancy increases, our proposed design performs better than the other two designs, and this performance gap increases with increasing discrepancy. This is because both the full factorial and the "pure computer model" designs lack the space-filling points that are critical in estimating the non linear discrepancy.

4 A Real Example

We apply our design strategy to a real example from P&G. The model is based on first principles, and involves a transformation of the P&G diaper line, where absorbent gelling material is being applied to a substrate. The example has been slightly modified for the benefit of simplicity

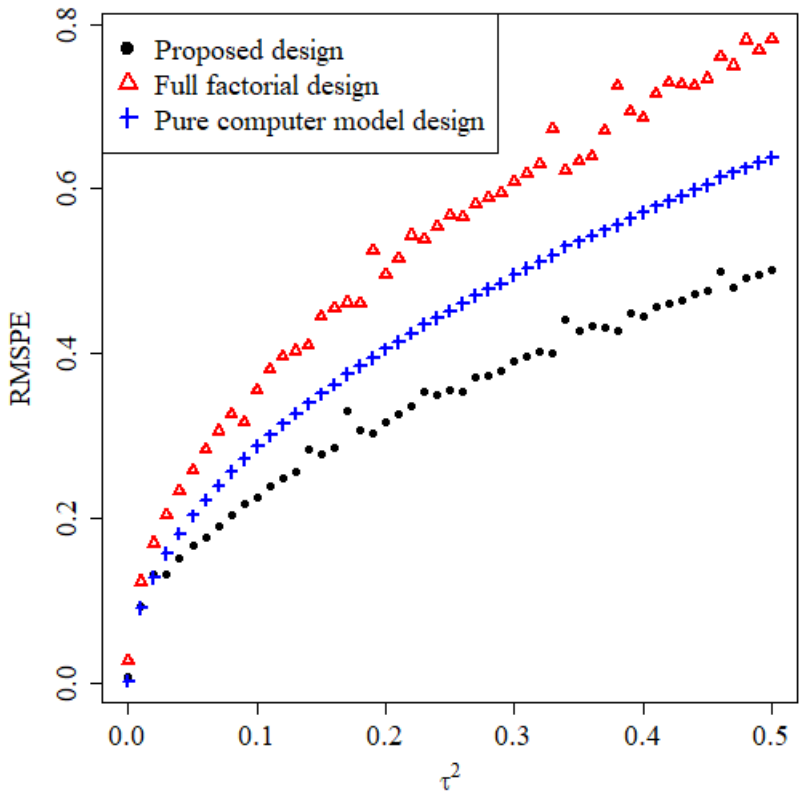

Figure 6: Comparing performance of our proposed design with increasing non-linear model discrepancy.

and to prevent disclosure of any potential sensitive information. The anonymized physics-based model is:

$$f(\boldsymbol{x};\boldsymbol{\eta}) = AGM \times B \times 2000, \tag{11}$$

where:

$$AGM = \left\{\frac{(10^3 x_2)^{2\eta_3} {x_5}^{2\eta_3-1}}{\eta_1^2 (c_4)^{2\eta_3-1}} + 10^6 k_2 x_1 x_2\right\}\left\{(e^{x_4} - c_1)c_2 + c_3\right\}x_5,$$

$$B = 1 - \left\{k_1 - \left(\frac{10^{-3}A}{(e^{x_4} - c_1)c_2 + c_3}\right)\frac{1}{x_5} - x_3\right\}^2 \frac{1}{\eta_2},$$

where the controllable process variables are $\boldsymbol{x} = \{x_1, ..., x_5\}$ and the unknown calibration parameters $\boldsymbol{\eta} = \{\eta_1, ..., \eta_3\}$. The details of the variables are omitted for confidentiality reasons. The budget to calibrate the physics-based model is assumed to be $N = 16$ points.

P&G has provided us with 646 physical experimental data points gathered for multiple pur-

poses including calibration of the physics-based model. The unknown values of the three calibration parameters were estimated from this dataset as $\boldsymbol{\eta}_0 = (1.5, 200, 0.2)'$. Figure 7 (left) plots the physical experiment data against the predictions from the calibrated model, which shows a good agreement between the two. Therefore, we will take this calibrated model as the "true" model and use it to evaluate the proposed design with the existing designs.

Suppose we have a budget for $N = 16$ runs. No information about the confidence in the computer model is assumed. Therefore, we choose $\gamma = 2q/N = 6/N$. This leads to $n = [\gamma N] = 6$, which can be used for efficiently estimating the calibration parameters. The remaining 10 runs can be used for estimating the model discrepancy. Assume the following prior distributions for the calibration parameters: $\eta_1 \sim \mathcal{U}[0.1, 10], \eta_2 \sim \mathcal{U}[0.1, 1000]$, and $\eta_3 \sim \mathcal{U}[0, 0.4]$. Since the computer model is cheap-to-evaluate, there is no need to incorporate any surrogate model uncertainties. So, we use Algorithm 2 to generate the proposed design.

We will compare the proposed design with (a) "pure computer model" design and (b) Maximin augmented nested Latin hypercube design (MmANLHD) proposed by Leatherman et al. (2017). Since the computer model is cheap-to-evaluate, we can view the LHD used for approximating the computer model to have an infinite number of runs and therefore, the MmANLHD reduces to a maximin design. It is easy to see that the maximin design in 16 runs is a 2^{5-1} maximum resolution fractional factorial design. For generating the "pure computer model" design, we use Algorithm 2 with $\gamma = 1$.

For each of the experimental designs, the output is simulated as:

$$y_i = f(\boldsymbol{x}_i; \boldsymbol{\eta}_0) + \epsilon_i, \tag{12}$$

where $\epsilon_i \overset{i.i.d}{\sim} \mathcal{N}(0, \sigma^2)$ with $\sigma^2 = 3$. We follow the same model fitting method described in Section 3 except that we do not include the model discrepancy.

The absolute prediction error is computed on each of the 646 points in the real dataset using the calibrated model for each of the designs. Figure 7 (right) compares the distribution of absolute prediction errors of the competing "pure computer model" design and maximin design, with our proposed design. We observe that the "pure computer model" design corresponds to the least prediction error. This is because there is no discrepancy in the computer model as seen in Figure 7 (left). Since all the 16 runs of the "pure computer model" design are based on D-optimal design

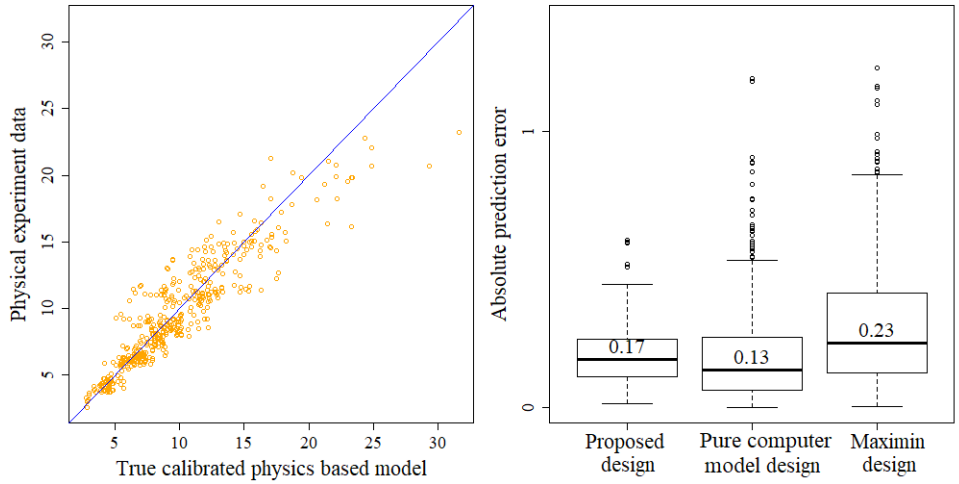

Figure 7: (left): Physical experiment data vs the true calibrated physics-based model from P&G; (right): Distribution of the absolute prediction error ratio in case of an unbiased physics-based model.

points, it provides the most accurate estimate of the calibration parameters, and thereby the most accurate calibration. As our proposed design has six runs of the D-optimal design points, it does better than the maximin design.

We are particularly interested to see the performance of our proposed design in the presence of model discrepancy. As the computer model $f(\boldsymbol{x};\boldsymbol{\eta})$ is unbiased, we will add a randomly generated realization of a Gaussian Process discrepancy to the true model to simulate the physical experiment output. Instead of simulating the physical experiment output using (12), it is simulated as:

$$y = f(\boldsymbol{x}_i;\boldsymbol{\eta}_0) + \delta(\boldsymbol{x}_i) + \epsilon_i, \tag{13}$$

where $\epsilon_i \overset{i.i.d}{\sim} \mathcal{N}(0,3)$, and $\delta(\boldsymbol{x})$ is a random realization from $GP(0,\tau^2 R(\cdot))$ with $\tau^2 = 6$, and $R(\boldsymbol{h}) = \exp(-\|\boldsymbol{h}\|^2)$. We fit the model in (1) on the simulated output to estimate $\boldsymbol{\eta}$ and the model discrepancy $\delta(\boldsymbol{x})$. The same method of estimation, as in Section 3, is used. The calibrated model is given by (10).

Figure 8 (right) plots the absolute prediction errors of the three designs. We observe that our proposed design now corresponds to the least prediction error, which shows that it provides the most accurately calibrated model. A very drastic drop in the relative performance is observed in the case of the "pure computer model" design, as it does not have any points to estimate model discrepancy. On the other hand, the maximin design does not use any information from the computer

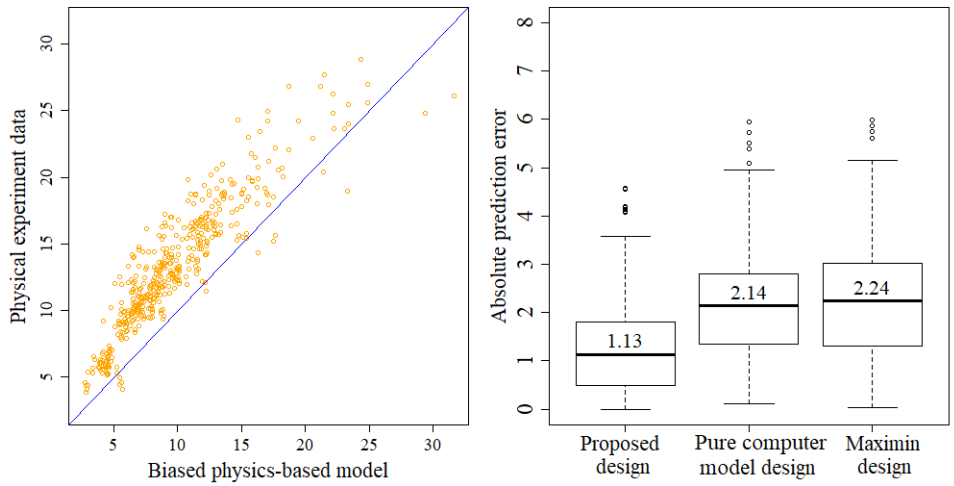

Figure 8: (left): Physical experiment output vs the biased physics-based model output; (right): Distribution of the absolute prediction error ratio in case of a biased physics-based model.

model. So, it does not have optimal points to estimate the calibration parameters, which leads to a sub-optimal performance. In contrast, our proposed design uses the information from the computer model, while also accounting for the model discrepancy, which leads to a better overall performance than both the competing designs.

5 Conclusion

This article presented a strategy for designing physical experiments when a computer model is available to the experimenter. Optimal designs can be generated by directly using the computer model, but such designs are susceptible to possible model violations. Our design strategy augments the optimal design points with space-filling points. These space-filling points act as check points for the computer model and protect against possible model violations. The proposed designs can become inferior to the optimal designs when the computer model is perfect but will be superior when the computer model is imperfect. Since the optimal designs are a subset of our proposed design, the loss of efficiency when the computer model is perfect is minimal, and for that reason we claim that our designs are model-robust.

The proposed design strategy is simple and also flexible to be extended. Our strategy can be modified to augment points when some information of the model discrepancy is available. For example, adding points corresponding to the maximum and minimum of the computer model can

be used to efficiently estimate a location-scale bias of the computer model. These can be viewed as "features" of the computer model that can act as useful model validation points. In fact, this also suggests that experimenters can extract other features from the computer model, such as all the local maxima and minima, and add them to the design, even when the connection to a discrepancy model is not evident. We hope to investigate this further in a future work.

About the authors

Arvind Krishna is a PhD candidate in the Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology, Atlanta. His email address is akrishna39@gatech.edu.

V. Roshan Joseph is a A. Russell Chandler III Professor in the Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology, Atlanta. He is a Fellow of ASQ. His email address is roshan@gatech.edu.

Shan Ba is a data science applied researcher at LinkedIn. His email address is shan.ba@linkedin.com.

William A. Brenneman is a Research Fellow and the Global Statistics Discipline Leader at Procter & Gamble and an Adjunct Professor of Practice in the Stewart School of Industrial and Systems Engineering at the Georgia Institute of Technology. He is a Fellow of ASQ. His email address is brenneman.wa@pg.com.

William R. Myers is a Principal Statistician at The Procter & Gamble Company. His email address is myers.wr@pg.com.

Acknowledgements

This research is supported by the U.S. National Science Foundation grants DMS-1712642 and CMMI-1921646.

References

Arendt, P. D., Apley, D. W. & Chen, W. (2016), 'A preposterior analysis to predict identifiability in the experimental calibration of computer models', *IIE Transactions* **48**(1), 75–88.

Ba, S. & Joseph, V. R. (2018), 'Maxpro: Maximum projection designs. R package version 4.1-2', *URL: https://cran.r-project.org/web/packages/MaxPro* .

Box, G. E. P. & Draper, N. R. (1959), 'A basis for the selection of a response surface design', *Journal of the American Statistical Association* **54**(287), 622–654.

Box, G. E. P. & Hunter, W. G. (1962), 'A useful method for model building', *Technometrics* **4**, 301–318.

Chaloner, K. & Verdinelli, I. (1995), 'Bayesian experimental design: A review', *Statistical Science* pp. 273–304.

Chang, Y. J. & Notz, W. I. (1996), 'Model robust designs', *Handbook of statistics* **13**, 1055–1098.

Christophe, D. & Petr, S. (2019), 'Randtoolbox: Generating and testing random numbers. R package version 1.30.0', *URL: https://cran.r-project.org/web/packages/randtoolbox* .

Dror, H. A. & Steinberg, D. M. (2006), 'Robust experimental design for multivariate generalized linear models', *Technometrics* **48**(4), 520–529.

Huang, C., Joseph, V. R. & Mak, S. (2020), 'Population quasi-monte carlo', *arXiv preprint arXiv:2012.13769* .

Johnson, M. E., Moore, L. M. & Ylvisaker, D. (1990), 'Minimax and maximin distance designs', *Journal of statistical planning and inference* **26**(2), 131–148.

Joseph, V. R. (2016), 'Space-filling designs for computer experiments: A review', *Quality Engineering* **28**(1), 28–35.

Joseph, V. R., Gu, L., Ba, S. & Myers, W. R. (2019), 'Space-filling designs for robustness experiments', *Technometrics* **61**, 24–37.

Joseph, V. R., Gul, E. & Ba, S. (2015), 'Maximum projection designs for computer experiments', *Biometrika* **102**(2), 371–380.

Joseph, V. R., Gul, E. & Ba, S. (2019), 'Designing computer experiments with multiple types of factors: The maxpro approach', *Journal of Quality Technology* pp. 1–12.

Joseph, V. R. & Melkote, S. N. (2009), 'Statistical adjustments to engineering models', *Journal of Quality Technology* **41**(4), 362–375.

Kennedy, M. C. & O'Hagan, A. (2001), 'Bayesian calibration of computer models', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63**(3), 425–464.

Kiefer, J. (1985), *Collected Papers III: Design of Experiments*, Springer.

Leatherman, E. R., Dean, A. M. & Santner, T. J. (2017), 'Designing combined physical and computer experiments to maximize prediction accuracy', *Computational Statistics & Data Analysis* **113**, 346–362.

Mak, S. (2019), 'Support points. R package version 0.1.4', *URL: https://cran.r-project.org/src/contrib/Archive/support* .

Mak, S. & Joseph, V. R. (2018), 'Support points', *The Annals of Statistics* **46**(6A), 2562–2592.

Masoudi, E., Holling, H., Duarte, B. P. & Wong, W. K. (2019), 'A metaheuristic adaptive cubature based algorithm to find bayesian optimal designs for nonlinear models', *Journal of Computational and Graphical Statistics* **28**(4), 861–876.

Masoudi, E., Holling, H., Wong, W. K. & Kim, S. (2020), 'Icaod: An r package for finding optimal designs for nonlinear models using imperialist competitive algorithm. R package version 1.0.1', *URL: https://cran.r-project.org/web/packages/ICAOD* .

Plumlee, M. (2017), 'Bayesian calibration of inexact computer models', *Journal of the American Statistical Association* **112**(519), 1274–1285.

Pukelsheim, F. & Rieder, S. (1992), 'Efficient rounding of approximate designs', *Biometrika* **79**(4), 763–770.

Qian, P. Z. (2009), 'Nested latin hypercube designs', *Biometrika* **96**(4), 957–970.

Ranjan, P., Lu, W., Bingham, D., Reese, S., Williams, B. J., Chou, C.-C., Doss, F., Grosskopf, M. & Holloway, J. P. (2011), 'Follow-up experimental designs for computer models and physical processes', *Journal of Statistical Theory and Practice* **5**(1), 119–136.

Sacks, J., Welch, W. J., Mitchell, T. J. & Wynn, H. P. (1989), 'Design and analysis of computer experiments', *Statistical science* pp. 409–423.

Santner, T. J., Williams, B. J. & Notz, W. I. (2018), *The design and analysis of computer experiments*, Springer.

Silvey, S. (1980), *Optimal design: an introduction to the theory for parameter estimation*, Vol. 1, Chapman & Hall.

Tuo, R. & Wu, C. F. J. (2015), 'Efficient calibration for imperfect computer models', *The Annals of Statistics* **43**(6), 2331–2352.

Williams, B. J., Loeppky, J. L., Moore, L. M. & Macklem, M. S. (2011), 'Batch sequential design to achieve predictive maturity with calibrated computer models', *Reliability Engineering & System Safety* **96**(9), 1208–1219.

Wu, C. F. J. & Hamada, M. S. (2011), *Experiments: Planning, Analysis, and Optimization*, Vol. 552, John Wiley & Sons.