Varying coefficient models and design choice for Bayes linear emulation of complex computer models with limited model evaluations

Varying coefficient models and design choice for Bayes linear emulation of complex computer models with limited model evaluations *

Amy L. Wilson †, Michael Goldstein ‡, and Chris J. Dent*

Abstract. Computer models are widely used to help make decisions about real-world systems. As computer models of large and complex systems can have long run-times and high-dimensional input spaces it is often necessary to use emulation to assess uncertainties in computer model output. This paper presents methodology for emulation of complex computer models motivated by a real-world example in energy policy. The computer model studied is an economic model of investment in electricity generation in Great Britain. The computer model was used to select parameters in a government policy designed to incentivise investment in renewable technologies to meet government targets. Limited computing time meant that few runs of the computer model were available to fit an emulator. The statistical methodology developed was therefore focussed on accurately capturing the uncertainty in computer model output arising from the small number of available model runs. A varying coefficient emulator is proposed to model uncertainty in model output when extrapolating away from model runs. To maximise use of the small number of runs available, this varying coefficient emulator is paired with a criterion-based procedure for design selection.

Key word. Uncertainty analysis, computer models, emulation, Bayes linear analysis, energy systems

AMS subject classifications. 62P30, 62F15, 60G15

1. Introduction.

Computer models of complex systems are used in many fields to help make decisions about these systems. These computer models combine a set of input assumptions with some approximation of a system to give some output of interest. A computer model can therefore be thought of as a function $\mathbf{f}(\cdot)$, taking a vector of inputs $\mathbf{x}$ and returning a vector of outputs $\mathbf{f}(\mathbf{x})$. It is usually the case that the modeller is interested in the behaviour of the real-world system, rather than the output of the computer model, making it necessary to study the uncertainties that link the model to this real-world system. Uncertainties that should be considered include: parametric uncertainty, arising from lack of knowledge about the appropriate input parameters ($\mathbf{x}$) to use, and structural discrepancy, which relates to the imperfect approximation of the system (by $\mathbf{f}(\cdot)$).

For fast computer models with small input spaces, parametric uncertainty can be assessed using standard Monte Carlo simulation and calibration can be performed using Markov Chain Monte Carlo techniques. For slow computer models or models with a large input space, these methods become computationally infeasible. To resolve this issue, emulators (statistical models of computer models) are commonly used to quantify the uncertainty in the output of a computer model at an untested input. For a given input $\mathbf{x}$, an emulator gives a probability distribution for the value of $\mathbf{f}(\mathbf{x})$. An emulator can be combined with Monte Carlo simulation over the joint distribution of $\mathbf{x}$ to assess parametric uncertainty ([29], [22]) and can be used to calibrate a computer model against historical data using Bayesian techniques ([5], [20], [19]).

*Submitted to the editors.

Funding: This work was supported by EPSRC grants EP/K03832X/1 and EP/K036211/1.

†School of Mathematics, University of Edinburgh, Edinburgh, UK (Amy.L.Wilson@ed.ac.uk, Chris.Dent@ed.ac.uk).

‡Department of Mathematical Sciences, Durham University, Durham, UK (Michael.Goldstein@durham.ac.uk).

A common form for an emulator is to model the i-th element of the computer model output at input $\mathbf{x}$ as

$$f_i(\mathbf{x}) = \sum_{j=1}^{p_i} \beta_{ij} h_{ij}(\mathbf{x}) + \epsilon_i(\mathbf{x}), \tag{1.1}$$

where $B = \{\beta_{ij}\}$ are a set of unknown constants associated with a set of known and deterministic basis functions $h_{ij}(\mathbf{x})$ and $\epsilon_i(\mathbf{x})$ is a stochastic process uncorrelated with B with zero mean and covariance function $\sigma_i^2 c_i(\mathbf{x}, \mathbf{x}')$, where $c_i(\mathbf{x}, \mathbf{x}')$ depends only on $\|\mathbf{x}-\mathbf{x}'\|$ and gives a positive semi-definite covariance matrix. The term $\sum_{j=1}^{p_i} \beta_{ij} h_{ij}(\mathbf{x})$ is known as the mean function. There are often natural linear relationships between the inputs and outputs of a computer model. Specification of a mean function with regression terms as in (1.1) allows for these linear relationships to be directly modelled and for any prior information about the relationships to be incorporated. In a traditional Bayesian analysis, a common choice for $\epsilon_i(\mathbf{x})$ is a Gaussian Process model (for example, see [29], [30], [3]) combined with a prior distribution over the β_{ij}, σ_i^2 and any parameters in the correlation function. In a Bayes linear analysis ([15]) it is only necessary to specify prior means and covariances for β_{ij} and for $\epsilon_i(\mathbf{x})$ rather than complete probability distributions. See [5], [6], [8] and [37] for further details. For both Bayes linear and traditional Bayesian fitting, prior judgments are combined with model (1.1) and data consisting of N computer model evaluations $D = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)\}$ (where $\mathbf{y}_k = \mathbf{f}(\mathbf{x}_k)$ for $k \in \{1, \ldots, N\}$) to give updated beliefs as to the value of $\mathbf{f}(\mathbf{x})$ for any $\mathbf{x}$.

This article presents methodology for applications where the number of possible model evaluations N is small, relative to the number of inputs. The focus is on problems where there is sparse coverage of the input space so extrapolation outside the design is needed but the number of runs is still sufficient to fit an emulator (unlike, say, very large physical models). For very large and slow models where very few model runs are possible one option is to use a multi-level modelling approach [7]. This can be done by constructing a fast approximate version of the model (e.g. by coarsening the grid, changing the time step or approximating the solution) and building an emulator of this fast version of the model. This emulator can then be used as an informed prior for the emulator of the full slow model. Where a multi-level modelling approach is used the methodology presented here is still of use because there is still a need for careful extrapolation to avoid overconfidence when emulating the slow model.

The development of the methodology in this paper was motivated by a real-world example, concerning the need to make government policy decisions under uncertainty using a computer model of the long-term GB electricity supply. In this example, the objective was to set the parameters of a support scheme for renewable generation to give the best chance of meeting future government climate and cost targets. Computing limitations meant that it was only possible to perform 80 evaluations of this computer model to fit an emulator with fourteen input parameters. With limited model evaluations available to fit the emulator it was critical to make best use of these model evaluations. Two aspects of this problem will be focussed on in this paper. The first aspect relates to the form of the emulator. An adaptation of (1.1) with varying coefficients was used to capture uncertainty about the coefficients β_{ij} in regions where there were few model evaluations. The second aspect focuses on the iterative selection of model evaluations for fitting the emulator. By carefully choosing the experimental design, time is not wasted performing

model evaluations that are not ultimately of value. For the example in this paper, model evaluations were selected by minimising a specific design objective related to the problem under study.

The rest of this article is organised as follows. Section 2 describes the background to the problem and the motivating example. Section 3 presents a varying coefficient model for emulation of a complex computer model, describes the process of fitting this model using Bayes linear methods and demonstrates use of this model on the electricity system example described above. Section 4 discusses some general principles for design choice with limited data and applies these principles to the example. Section 5 presents results from the analysis of the electricity system example, including a discussion of the use of emulation in making decisions on the choice of strike prices for renewable technologies.

2. Background.

2.1. Emulation in problems with limited model evaluations.

When fitting an emulator to a small number of model evaluations, the need to minimise the number of independent parameters in the model (to avoid overfitting) must be balanced with the need to reflect accurately the form of the response surface and its associated uncertainty. This is particularly problematic when the parameters β_{ij} are thought to vary in different regions of the input space, a common problem with large and complicated computer models. With a large design, errors arising from an inexact prior assumption that each β_{ij} is constant over the whole space can be corrected for in the updated beliefs about the stochastic process $\epsilon_i(\mathbf{x})$. With a small design there may not be enough model evaluations to override the inexact prior assumption and so the predictive ability of the emulator may be poor. Whilst interaction terms in the mean function of (1.1) can deal with the issue of coefficients which vary in different regions of the space to some extent, they can only be used for simple linear relationships. For large computer models, interactions are likely to be complex and non-linear. If it is thought that some or all of the β_{ij} may be non-constant over the input space then it may be possible to obtain a better fit to the computer model using an emulator which allows for coefficients to vary in different regions of the input space.

A further benefit of incorporating varying coefficients into the the emulator is to guard against the underestimation of uncertainty when extrapolating away from model evaluations. The emulator in (1.1) uses regression terms to model the global behaviour of the computer model. In regions of the space with few model evaluations, this fitted regression model is extrapolated. In contrast, an emulator which uses only the stochastic process term in (1.1) will revert to a mean of zero when extrapolating. Use of regression terms can therefore be helpful when computing time is limited because the fit of the emulator in regions with design points is extrapolated into sparse regions but there is a risk that uncertainty is underestimated. This underestimation of uncertainty arises because an assumption is made that the same polynomial mean function can be used to describe the global response surface of the computer model everywhere. In practice, for complex models, there is likely to be considerable uncertainty as to the form of the most appropriate polynomial global response surface in regions of the space with few model evaluations. A varying coefficient model explicitly models this uncertainty by allowing the coefficients of the regression terms to vary across the input space. By assuming that coefficients are constant, uncertainty in regions of the space with no model evaluations will be underestimated, as uncertainty in the form of the polynomial mean function is not accounted for. This is particularly a problem for small designs, which will have a sparse coverage of the input space, because

making decisions based on such designs will necessarily involve extrapolation. A varying coefficient model has a greater number of parameters to be fitted, which can be a challenge with a small design, but these extra parameters give the flexibility to introduce additional uncertainty when there is uncertainty as to the stability of the polynomial mean function across the input space. We argue that unless the prior information suggests that same polynomial mean function can be used across the relevant input space it is crucial to capture the increased uncertainty arising from lack of knowledge of this mean function in the modelling process.

The problem of varying coefficients can be dealt with by fitting different emulators to different regions of the input space. When using emulators to history match, model evaluations are performed in waves (e.g. see [37], [5], [6], [38]). After each wave, a new emulator is fitted and used to restrict the input space to inputs that could plausibly match system observations. Then a new wave of analysis is performed on this smaller input space. As a new emulator is fitted with each restriction of the input space, different prior assumptions for the β_{ij} (and also the covariance function) can be used. In [16],[21] and [32] partitioning methods are used to divide the input space into different regions. A different emulator is fitted within each of these regions, so the β_{ij} are not assumed to be constant over the input space (and the covariance function is not assumed to be stationary). These methods are very flexible but require enough model evaluations to fit a separate emulator in each region. Where model evaluations are limited, this can result in very high levels of emulator uncertainty over the entire input space. For complex computer models with very different response surfaces in different regions, this high level of uncertainty may be warranted, but in many cases the values of β_{ij} in one region of the space may be informative about the values of β_{ij} in neighbouring regions. Making use of this information can help reduce the emulator uncertainty over the whole input space.

Methods for Bayes linear estimation of parameters in general regression models without an assumption of a constant regression coefficient were presented in [13] and [14]. These general regression models were fitted over the whole dataset, allowing every datapoint to contribute to estimation of the coefficient, rather than by partitioning the variable space. In [31] a fully specified Bayesian formulation for a smoothly varying coefficient regression model was given in the context of choosing an optimal design. Variation in coefficients was modelled using a Gaussian process with specified covariance function. This covariance function determines the extent to which coefficients vary throughout the space. [12] extended this approach to use a varying coefficient model for spatial data. Further applications of this spatially varying coefficient model can be seen in [11] and [18]. In this paper, a varying coefficient model is proposed for emulation of complex computer models. A varying coefficient model explicitly models the uncertainty in the coefficients β_{ij}, which is of particular use when the number of model evaluations is limited because of the need to extrapolate outside the dataset.

An alternative approach when assumptions of constant mean and variance do not hold is to use a non-stationary Gaussian Process model ([35], [28]). In [1] a Gaussian Process is used to model the global mean function in addition to the local variation. Our approach differs in that we use a polynomial global mean function and associate a stochastic process with each polynomial term in this function. This makes it possible to incorporate any prior information about the coefficients β_{ij} into the analysis. In [39] a non-stationary Gaussian Process is formed from a mixture of stationary processes, where the weights determining the mixing depend on the dominant local behaviour of the computer model.

Our proposed model can also be thought of as a sum of stationary stochastic processes but we instead explicitly link each of these processes with one of the coefficients of the global mean function. This makes the prior specification more transparent and reduces the number of parameters that must be estimated (as no mixing weights are needed) which is an important consideration when few model runs are possible.

Our proposed emulator is tested on the motivating example of a complex computer model of the UK electricity system. Unlike the alternative approaches described above (e.g. [1] and [39]), the emulator is fitted using Bayes linear techniques ([15]), rather than using fully specified probability distributions. The advantage of using Bayes linear methods is that it is only necessary to specify prior expectations and covariances for quantities of interest rather than full probability distributions. The result of this simplification is that computations are reduced to linear algebra and so Markov Chain Monte Carlo methods are not required to fit the emulator. Therefore, design calculations can be carried out which would otherwise be computationally intractable.

2.2. Choice of design for emulation with limited model evaluations. In applications where it is only possible to evaluate the computer model for a limited selection of input parameters it is necessary to make a careful choice of these input parameters in order to maximise the use of each model run. There are many options for constructing an experimental design for a computer experiment ([34]). A common choice is to use a space-filling design such as a Latin Hypercube sample ([24], [25], [36], [23]) to give good coverage of the input space. Often in studies using computer models, the objective is not to fit an emulator, but to use the emulator to make some decision under uncertainty (e.g. to find optimal inputs or to find inputs that meet some given criterion or to history match). With this in mind, using a space-filling design when computing time is limited can be a poor choice because model evaluations are wasted on improving the emulator in regions of the input space that have little impact on the final objective of the study. Criterion based methods which account for the output of the computer model, particularly in combination with sequential choice of design points (individually or in waves), can be a better choice as model evaluations can be chosen specifically with the aim of the study in mind.

This paper presents a criterion based method for the sequential selection of model evaluations. Examples of criterion based methods for design selection can be seen in [2], [37], [7] and [17]. These methods all seek to reduce the variance of the emulator over the full input space, which is appropriate where all regions of the input space are equally important. For the motivating example of a computer model of the UK electricity system, the ultimate purpose of the study was to find model inputs that met key government targets. It was therefore important to focus the design on improving the emulator in regions of the input space with a high probability of meeting these targets, making minimisation of the emulator variance over the full space an inappropriate choice for a design criterion. The iterative method presented in this paper instead weights points in a grid at which the criterion is evaluated so that the design can be focussed on improving the emulator in regions of the input space that are relevant for the ultimate purpose of the study.

2.3. Motivating example. The example studied in this paper is a computer model of the long-term UK electricity supply. The model is proprietary software used by industry and government to investigate energy policy and hence many of the technical details are confidential. This confidentiality does not impact our results as we are concerned with the statistical methodology used for emulation.

The model takes inputs such as projected demand, fossil fuel prices and the costs of future technology and uses these assumptions to model investment in generation and electricity supply in the UK. Investment decisions are based on projected future cash-flows by assessing whether a plant will exceed the user-specified rate of return required by investors. The computer model estimates wholesale electricity prices by comparing daily demand curves on sample days (net of wind generation, interconnection, storage and reserve requirements) to the generation merit order. This merit order is estimated using assumptions about the short term costs for each plant in the system. The projected income of a plant can be estimated from the projected wholesale electricity prices and the merit order (which determines whether a plant will be used on a particular day). Projected costs are estimated using user assumptions about the set-up and running costs associated with different types of generating plant. Outputs from the computer model are wide-ranging and include future costs, generation mix and emissions.

The computer model is used to study aspects of government energy policy such as the amount of generation to incentivise to reduce the risk of shortfalls in electricity supply. In this paper, we focus on one particular application, which was to study a new policy for providing support for renewable generation in relation to three outputs: the projected cost of government support in 2020, the proportion of UK electricity generation provided by renewable technology in 2020 and the level of CO2 emissions in 2030. The policy was part of the UK government Electricity Market Reform program ([9]) and aimed to reform the system for making payments to large scale renewable generators whilst still incentivising investment into these technologies. Under the new policy, potential renewable generation projects can bid for a contract for difference at a specific 'strike price'. If the project is successful in this bid, then the generator is guaranteed to receive this strike price for electricity produced over the next 15 years. If the price of electricity is lower than the strike price, then the government will make up the difference, and if the price of electricity is higher than the strike price then the generator must re-pay the difference ([10]).

The computer model was used to assess which strike prices would likely result in a total cost in 2020–21 of less than £7.6 bn, a proportion of electricity provided by renewable generation of at least 30% in 2020 and CO2 emissions of less than 100 gCO2/kWh in 2030 ([27]). The restriction on cost arises from the Levy Control Framework ([26]), whilst the restrictions on renewable generation and emissions are set to comply with EU targets. A different strike price is used for each renewable technology as technologies which are less well developed require a larger price to offset risk, and strike prices are expected to reduce through time, as the cost of developing new technology reduces.

As well as modelling the effect of different strike prices on computer model outputs, the computer model was used to assess the effect that different assumptions about the future electricity system have on these outputs. Strike prices are under the control of the government, but assumptions such as future electricity demand and future fuel prices are not, and so uncertainty in these inputs should be modelled to get an accurate picture of the effect that changes in these assumptions will have on the targets. In this paper focus is given to eight uncertain assumptions: electricity demand, coal price, gas price, the construction costs of different plants, the load factors for onshore and offshore wind power (the ratio of actual output to theoretical maximum output) and the hurdle rates for onshore and offshore wind power (the rate of return required by investors in these technologies). These inputs were chosen after discussion with expert modellers because they were thought to have a large impact on computer model outputs and were also highly uncertain.

In this example, the use of emulation is investigated as a methodology for identifying strike prices which meet government cost, renewable generation and emissions objectives with a high probability, accounting for uncertainty in parameters and model discrepancy. As the model is proprietary software, access to the model and ability to make changes to speed-up the model run-time was restricted. This, combined with the one-hour run-time meant that only 80 model runs were possible. Use of proprietary software is not unusual in the energy policy field and it is often the case that there is limited time available for analysis in practice as decisions are made quickly in response to policy changes. Thus methodology is needed that can maximise use of limited data. This was especially necessary for the motivating example because the effect of different model inputs was thought to vary in different regions of the parameter space.

3. Emulation.

3.1. Varying coefficient emulator.

Let $\mathbf{f}(\mathbf{x})$ be the output of a deterministic computer model at some vector of inputs $\mathbf{x}$. The precise value of $\mathbf{f}(\mathbf{x})$ is unknown at untested $\mathbf{x}$ so we represent our uncertainty in the i-th dimension of $\mathbf{f}(\mathbf{x})$ as

$$f_i(\mathbf{x}) = \sum_{j=0}^{p_i} \beta_{ij} h_{ij}(\mathbf{x}) + \sum_{j=0}^{p_i} \epsilon_{\beta_{ij}}(\mathbf{x}) h_{ij}(\mathbf{x}), \tag{3.1}$$

where $h_{ij}(\mathbf{x})$ are a set of known and deterministic basis functions with $h_{i0}(\mathbf{x}) = 1$, $B = \{\beta_{ij}\}$ are unknown constants and $\epsilon_{\beta_{ij}}(\mathbf{x})$ are a set of stochastic processes. The $\epsilon_{\beta_{ij}}(\mathbf{x})$ are assumed to be uncorrelated with B and each other and to have zero mean and covariance $\sigma_{ij}^2 c_{ij}(\mathbf{x}, \mathbf{x}')$, where $c_{ij}(\mathbf{x}, \mathbf{x}')$ is some correlation function that is dependent only on $\|\mathbf{x}-\mathbf{x}'\|$. A common choice for the correlation function is the Gaussian correlation function, given by

$$c_{ij}(\mathbf{x}, \mathbf{x}') = \exp\left(- \sum_k \frac{(x_k - x_k')^2}{\delta_{ijk}^2} \right), \tag{3.2}$$

where the δ_{ijk} are constants to be specified. Note that the correlation lengths δ_{ijk} are allowed to vary with the input dimension. This can be useful if certain inputs are thought to have a bigger effect on the correlation but in practice may have little effect on the results (see the motivating example for further discussion).

We use Bayes linear methods ([15]) to fit (3.1). Bayes linear methods only require prior means and covariances for quantities of interest (rather than full probability distributions). There are several resulting advantages. Firstly, it is not necessary to elicit judgments of the precise forms of prior distributions for emulator parameters from experts. This is particularly useful because the effect of some emulator parameters may be non-intuitive and so eliciting accurate prior judgments can be difficult. Secondly, regardless of the choice of priors, Markov Chain Monte Carlo (MCMC) methods are not required for the updating of prior beliefs. Without the need for MCMC, calculations are simplified and sped up, making it possible to perform more complicated analyses using these updated beliefs. One such example is seen in section 4, where the effect of adding different potential design points to a design is evaluated against a criterion. For each potential design point, updated beliefs for the emulator parameters based on this extra design point must be obtained to evaluate the criterion. Using Bayes Linear methods rather than MCMC reduces the computational burden as the time spent fitting the emulator is reduced. It would also be possible to fit

(3.1) using other methods, for example a traditional Bayesian approach using Gaussian Processes – this is discussed further in subsection 3.2.

In this paper, the dimensions of $\mathbf{f}(\mathbf{x})$ are assumed to be independent. It is theoretically possible to fit a multivariate emulator which accounts for dependence between dimensions (e.g. [33], [4]) but in the motivating example studied the mean function adequately accounted for any dependence between dimensions. To complete the Bayes linear prior specification, prior judgments must be made for $\mathbb{E}[\beta_{ij}]$ and $\text{Cov}[\beta_{ij}, \beta_{ik}]$ for all i, j.

Model (3.1) can be written equivalently as

$$f_i(\mathbf{x}) = \sum_{j=1}^{p_i} \epsilon'_{\beta_{ij}}(\mathbf{x}) h_{ij}(\mathbf{x}) + \epsilon'_{\beta_{i0}}(\mathbf{x}), \tag{3.3}$$

where $\epsilon'_{\beta_{ij}}(\mathbf{x}) = \beta_{ij} + \epsilon_{\beta_{ij}}(\mathbf{x})$ for $j \in \{1, \ldots, p_i\}$. Comparing (3.3) with (1.1), we see that the parameters governing the relationship between the basis functions and the output in (3.3) are allowed to vary as stochastic processes. These stochastic processes are dependent on the inputs of the computer model and so vary across the input space. The extent of the variation over the input space is governed by the correlation function of each stochastic process. If $\epsilon'_{\beta_{ij}}(\mathbf{x})$ is highly correlated over the input space then its value will not vary much with different inputs, and so data in all regions of the space can contribute to the estimation of this value (when the correlation of $\epsilon'_{\beta_{ij}}(\mathbf{x})$ and $\epsilon'_{\beta_{ij}}(\mathbf{x}')$ is one for all i and all $j > 0$ and all pairs $\mathbf{x}$ and $\mathbf{x}'$, (3.3) reduces to (1.1)). Conversely, if $\epsilon'_{\beta_{ij}}(\mathbf{x})$ has a low correlation over the input space, then its value will vary in different regions of the space, and so more data over the whole region will be required to reduce the uncertainty associated with its value. The parameters of the correlation function can therefore be used as tuning parameters to determine the extent to which information is borrowed across the space. For small datasets a careful prior choice of these tuning parameters is needed to ensure sufficient flexibility of the emulator whilst extracting maximum information from the limited dataset.

As discussed in section 2, the additional stochastic processes associated with the coefficients β_{ij} in (3.1) explicitly model the uncertainty in the form of the polynomial mean function given by the term $\sum_{j=0}^{p_i} \beta_{ij} h_{ij}(\mathbf{x})$. For small datasets it will be necessary to extrapolate. Any extrapolation is inherently assumption based but in order not to be overconfident when extrapolating it is important to ground these assumptions in the scientific understanding of the problem. For example, it may be that there are scientific reasons to believe that a linear regression should fit well over the whole parameter space. If the regression parameters accord with this scientific understanding then it is often reasonable to be directed by this regression when extrapolating. Where this is not the case, our proposed approach in (3.1) offers the flexibility to compromise between strictly following the form of the polynomial mean function or ignoring it and simply inflating the predictive variance. Away from data, the mean of (3.1) will revert to the polynomial mean function (the same as (1.1)) but the uncertainty will be greater than (1.1) because of the additional stochastic process terms $\epsilon_{\beta_{ij}}(\mathbf{x})$. An alternative approach might be to increase the prior variance associated with the β_{ij} in (1.1) but this would overinflate the variance in all regions of the space (i.e. not just far from data) and would also mean that if there were lots of model evaluations in one region of the input space, the variance of the coefficients would reduce over the whole space, failing to capture the extrapolative uncertainty. In general we would also recommend testing any assumptions made when extrapolating using a small test set of further model evaluations at values far from existing data where the polynomial mean

function gave very different predictions to those close to existing evaluations (note that due to access to the model this was not possible with the motivating example).

3.2. Fitting the emulator - Bayes linear analysis. The Bayes linear updating equations are used to update prior beliefs about the emulator, given data D. For a complex computer model, the data associated with the i-th element of the computer model output are N model evaluations so that $D_i = \{(\mathbf{x}_n, y_{in}),\ \text{for } n \in \{1, \dots\ N\}\}$, where $y_{in} = f_i(\mathbf{x}_n)$. The Bayes linear equation for the updated expectation of the i-th element of the computer model output at $\mathbf{x}$ is

$$\mathbb{E}_{D_i}[f_i(\mathbf{x})] = \mathbb{E}[f_i(\mathbf{x})] + \mathrm{Cov}[f_i(\mathbf{x}), D_i](\mathrm{Var}[D_i])^{-1}(Y_i - \mathbb{E}[D_i]), \tag{3.4}$$

where $Y_i = (y_{i1}, \dots, y_{iN})$ is a vector of the i-th element of the observed outputs. Expectations and variances in (3.4) are taken with respect to the prior judgments. Similarly, the Bayes linear equation for the updated covariance between two outputs $f_i(\mathbf{x})$ and $f_i(\mathbf{x}')$ is

$$\mathrm{Cov}_{D_i}[f_i(\mathbf{x}), f_i(\mathbf{x}')] = k(\mathbf{x}, \mathbf{x}') - \mathrm{Cov}[f_i(\mathbf{x}), D_i](\mathrm{Var}[D_i])^{-1}\mathrm{Cov}[D_i, f_i(\mathbf{x}')], \tag{3.5}$$

where $k(\mathbf{x}, \mathbf{x}')$ is the prior covariance between $f_i(\mathbf{x})$ and $f_i(\mathbf{x}')$.

Taking the expectation of (3.1), the expectation $\mathbb{E}[f_i(\mathbf{x})]$ in (3.4) is given by

$$\mathbb{E}[f_i(\mathbf{x})] = \sum_{j=0}^{p_i} \mathbb{E}[\beta_{ij}]h_{ij}(\mathbf{x}) + \sum_{j=0}^{p_i} \mathbb{E}[\epsilon_{\beta_{ij}}(\mathbf{x})]h_{ij}(\mathbf{x}) = \sum_{j=0}^{p_i} \mathbb{E}[\beta_{ij}]h_{ij}(\mathbf{x}) \tag{3.6}$$

and the covariance $k(\mathbf{x}, \mathbf{x}')$ in (3.5) is given by

$$\begin{aligned}\mathrm{Cov}[f_i(\mathbf{x}), f_i(\mathbf{x}')] = k(\mathbf{x}, \mathbf{x}') = &\sum_{j=0}^{p_i}\sum_{k=0}^{p_i} h_{ij}(\mathbf{x})h_{ik}(\mathbf{x}')\mathrm{Cov}[\beta_{ij}, \beta_{ik}] + \\ &\sum_{j=0}^{p_i} h_{ij}(\mathbf{x})h_{ij}(\mathbf{x}')\sigma_{ij}^2 c_{ij}(\mathbf{x}, \mathbf{x}').\end{aligned} \tag{3.7}$$

The vector $\mathrm{Cov}[f_i(\mathbf{x}), D_i]$ in (3.4) and (3.5) is given by $(k(\mathbf{x}, \mathbf{x}_{i1}), \dots, k(\mathbf{x}, \mathbf{x}_{iN}))^T$, and the matrix $\mathrm{Var}[D_i]$ is given by the matrix with (n, m)-th entry $k(\mathbf{x}_n, \mathbf{x}_m)$. By substituting these quantities into (3.4) and (3.5), adjusted beliefs for the expectation and variance of $f_i(\mathbf{x})$ for some untested $\mathbf{x}$ can be found. The Bayes linear updating equations can be used in a similar way to update prior judgments for β_{ij} and $\epsilon_{\beta_{ij}}(\mathbf{x})$, giving adjusted beliefs for these parameters for any $\mathbf{x}$, i and j.

After setting the prior beliefs and updating these beliefs using the Bayes linear equations, the fit of the emulator can be checked. One possible diagnostic is a leave-one-out plot, leaving each design point out in turn, fitting the emulator to the remaining points, and checking the emulator mean and variance for the left out point. If too many of the observed points lie outside the emulator prediction intervals then the model assumptions need investigating. In particular, the basis functions or correlation function may need adjusting. Validation of the emulator is discussed further in relation to the motivating example in subsection 5.1. A varying coefficient emulator is compared to a fixed coefficient emulator for three illustrative examples in section SM1 of the supplementary material.

As an alternative to the Bayes linear analysis set out above, we could instead assume that the stochastic processes $\epsilon_{\beta_{ij}}(\mathbf{x})$ in (3.1) are Gaussian Processes, specify full prior

distributions for the unknown parameters β_{ij}, σ_{ij}^2 and δ_{ijk} and fit the model using Bayesian updating by conditioning on the data D. After performing the Bayesian update (see [29]) the posterior distribution for unknown $f_i(\mathbf{x})$ is given by

$$f_i(\mathbf{x}) \mid D, \{\beta_{ij}\}, \{\sigma_{ij}^2\}, \{\delta_{ijk}\}) \sim \mathrm{GP}\left(m_i^*(\mathbf{x}), r_i^*(\mathbf{x}, \mathbf{x}')\right), \tag{3.8}$$

where if $X = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ are the inputs associated with D,

$$\begin{aligned}
m_i^*(\mathbf{x}) &= \sum_{j=0}^{p_i} \beta_{ij} h_{ij}(\mathbf{x}) + r_i(\mathbf{x}, X)^T R^{-1}(Y_i - m_i(X)), \\
r_i^*(\mathbf{x}, \mathbf{x}') &= r_i(\mathbf{x}, \mathbf{x}') - r_i(\mathbf{x}, X)^T R^{-1} r_i(\mathbf{x}, X), \\
m_i(X) &= \left(\sum_{j=0}^{p_i} \beta_{ij} h_{ij}(\mathbf{x}_1), \ldots, \sum_{j=0}^{p_i} \beta_{ij} h_{ij}(\mathbf{x}_N) \right) \\
r_i(\mathbf{x}, \mathbf{x}') &= \sum_{j=0}^{p_i} h_{ij}(\mathbf{x}) h_{ij}(\mathbf{x}') \sigma_{ij}^2 c_{ij}(\mathbf{x}, \mathbf{x}'), \\
r_i(\mathbf{x}, X)^T &= (r_i(\mathbf{x}, \mathbf{x}_1), \ldots, r_i(\mathbf{x}, \mathbf{x}_N)),
\end{aligned}$$

and R is given by the matrix with (n, m)-th entry $r_i(\mathbf{x}_n, \mathbf{x}_m)$. Markov Chain Monte Carlo in combination with this posterior distribution and prior distributions for β_{ij}, σ_{ij}^2 and δ_{ijk} could then be used to fit the emulator (e.g. see [19]).

3.3. Motivating example.

In this section, the emulator described in subsection 3.1 is fitted to an initial set of design points obtained from the energy policy computer model. Limited computing time was available to evaluate this model and so the varying coefficient emulator was used to accurately model uncertainty arising from sparse coverage of the input space.

3.3.1. Inputs and initial design.

In total, fourteen inputs to the computer model were included in the study. Six inputs were used to represent strike prices for three different renewable technologies (offshore wind power, onshore wind power and solar power) and eight inputs were used to model the uncertain parameters. We write the inputs of the computer model as $\mathbf{x} = (\theta, \mathbf{z})$, where θ represents the strike prices and $\mathbf{z}$ represents the remaining uncertain parameters.

The strike prices for each of the three technologies considered were represented by two parameters: the strike price in 2016, and the exponential rate of decay of the strike price through time to 2035. The other inputs varied in the study were: electricity demand (z_1), coal price (z_2), gas price (z_3), the construction costs of different plants (z_4), the load factors for offshore (z_5) and onshore (z_6) wind power and the hurdle rates for offshore (z_7) and onshore (z_8) wind power. The assumptions used in a government study ([27], [10]) of this problem were available and were used to parametrise each of these inputs. The electricity demand, coal price and gas price inputs (all time series) were represented as a shift away from the central government assumption. The construction costs, load factors and hurdle rates were represented as a multiple of the central government assumption. More details on the parametrisation of the inputs are given in section SM2 of the supplementary material.

Computer model evaluations had to be pre-prepared in sets off site, and so a stepwise procedure, selecting the next run based on results from the previous run, was not appropriate. As an initial design, a maximin Latin hypercube sample was used to select 40 design

points over the fourteen dimensional input space. For this initial design there was a fixed time available for model runs (one working week) and 40 was the number thought to be possible in that time. In practice, there was time to run an additional 16 design points. A second wave of analysis was completed later (again with a fixed period of access to the computer model), consisting of 24 design points, selected using a criterion-based design selection method. This method is discussed in section 4.

3.3.2. Emulator for initial design. Let $\mathbf{f}(\theta, \mathbf{z}) = (f_r(\theta, \mathbf{z}), f_e(\theta, \mathbf{z}), f_s(\theta, \mathbf{z}))$ be the computer model output, where the subscripts r, e and s represent the three one-dimensional outputs studied: renewable generation in 2020, emissions in 2030 and spend in 2020 respectively. The varying coefficient emulator given in (3.1) was used to model each dimension of $\mathbf{f}(\theta, \mathbf{z})$, with each of the three outputs assumed independent conditional on the input parameters. This is a reasonable assumption as the inputs were selected in discussion with experts to be those that are likely to impact the outputs so any residual dependence should be small.

By using a varying coefficient emulator, we can account for uncertainty in the coefficients β_{ij} when extrapolating outside the limited design. A varying coefficient emulator is also appropriate because the effects of different input parameters were thought to vary in different regions of the input space. The computer model determines whether different generating plants are constructed based on assumptions about the potential cashflows of these plants. Inputs relating to particular types of plant will have different impacts depending on whether these plants are constructed. For example, the load factor of an offshore wind plant will have more impact on the cost of support when a lot of offshore wind plants are constructed (where the number of plants constructed depends on the values of the other inputs). Initial fits of linear regression models to the design points supported this view because the estimated values of coefficients varied when different subsets of the design were used to fit the model. Later, in subsection 5.2, the fit of the varying coefficient model is compared to the fit of the fixed coefficient model using the full design.

Prior judgments and fitting of the emulator. Linear regression fits combined with expert judgment were used to select the vector of basis functions (both linear and non-linear terms) for each output. Table 1 gives the linear terms included in the basis function of each dimension of the emulator alongside the adjusted estimate of the coefficient β_{ij} for each term. The coefficient estimates reflect intuitive explanations for the relationships between the variables (this is discussed further in section SM4 of the supplementary material). Additional non-linear terms are listed beneath Table 1. Not all of the coefficients included in the model were allowed to vary and a different subset of the coefficients was allowed to vary for each of the three outputs. Selecting which subset of the coefficients in model (3.1) to vary was done with reference to the residual sum of squares when fitting the emulator, expert knowledge and the extent that coefficient estimates in linear regression fits varied when fitted to different subsets of the space. Note that the inputs to include in the model were determined before testing the extent to which the coefficients varied (so we did not select inputs on the basis of whether the coefficients were variable or not). Choosing which coefficients to vary based only on linear regression fits to subsets of the input space was not possible because the small size of the design meant that uncertainty in parameter estimates was large, and so it was difficult to distinguish between coefficients which might be varying and coefficients which were very uncertain.

The coefficients allowed to vary for each dimension of the emulator are listed in Table 2 along with prior judgments for the variance and correlation lengths for the stochastic

Input	Renewables	Emissions	Spend
1. Offshore strike price rate of decay	-0.14	0.13	-0.19
2. Offshore strike price starting price	0.35	-0.24	0.44
3. Onshore strike price rate of decay		-0.06	
4. Onshore strike price starting price		0.08	
5. Solar strike price rate of decay			
6. Solar strike price starting price			
7. Demand	-0.49	0.46	
8. Coal			
9. Gas		0.15	-0.44
10. Technology costs	-0.39	0.56	-0.29
11. Hurdle rate offshore	-0.12	0.16	-0.12
12. Hurdle rate onshore			
13. Load factor offshore	0.61	-0.42	0.57
14. Load factor onshore	0.22		

Table 1

Coefficient estimates for inputs included as linear terms in the basis functions for each emulator. Interaction terms also included for all three outputs were: (2,10,13), (2,10), (2,13), (10,13), where numbers correspond to the inputs listed in the table.

processes associated with these varying coefficients. These prior judgments were set by considering the expected size of variation of each coefficient, and the speed over which the coefficient might vary (in practice, these might be set in consultation with practitioners or based on previous studies). For the remainder of the prior assumptions, we set (for each dimension): $\mathbb{E}[\beta_j] = 0$; $\text{Cov}[\beta_j, \beta_k] = 0$ for $j \neq k$, $\text{Cov}[\beta_j, \beta_k] = 0.1$ for $j = k$ for the renewables and emissions emulators and $\text{Cov}[\beta_j, \beta_k] = 0.15$ for the emulator of cost. Some of these prior assumptions are simple, in particular those used for the coefficients β_j. For the motivating example it was not feasible to conduct a full expert elicitation exercise and the complex interactions between the various inputs made it difficult to determine in advance the effect that individual inputs would have, so we chose priors that reflected these limitations on our judgements.

The input and output data were scaled to lie between -1 and 1 before fitting the emulator. With the prior judgments given above, this scaling means that the prior variance (given in (3.7)) increases with the distance from the centre of the input space, so outliers will be associated with a larger prior variance.

Validation. To test the fit of the emulator, a leave-one-out cross validation was performed using the 56 initial design points. Each design point was removed in turn. The emulator was then fitted using the remaining 55 design points and used to predict the computer model output for the removed point. Figure 1 shows the results of this validation. Whilst the outputs for the majority of the design points are predicted well by the emulator, for around 15% of the design points in each dimension the computer model output is outside the probability interval associated with that design point, indicating that there may be an issue with the fit of the emulator. In total, 10 (renewables emulator), 7 (emissions emulator) and 11 (cost emulator) of the probability intervals in Figure 1 did not contain the true computer model output. Increasing the probability intervals to three standard deviations either side of the mean gives 2 (renewables emulator), 4 (emissions emulator) and 2 (cost emulator) of the true computer model outputs lying outside their prediction

Renewable model			Emissions model			Spend model		
Basis function	σ_j^2	δ_{jk}^*	Basis function	σ_j^2	δ_{jk}^*	Basis function	σ_j^2	δ_{jk}^*
Constant	0.15^2	$1, 3$	Constant	0.15^2	$1, 5$	Constant	0.175^2	$1, 5$
2	0.11^2	$4, 10$	10	0.18^2	$4, 10$	2	0.15^2	$4, 10$
$(2, 10, 13)$	0.11^2	$4, 10$	13	0.18^2	$4, 10$	$(2, 10, 13)$	0.15^2	$4, 10$
$(2, 10)$	0.11^2	$4, 10$				$(2, 10)$	0.15^2	$4, 10$
$(2, 13)$	0.11^2	$4, 10$				$(2, 13)$	0.15^2	$4, 10$
$(10, 13)$	0.11^2	$4, 10$						

Table 2

*Prior judgments for the covariance function associated with each basis function. The basis function number corresponds to numbers given for inputs in Table 1. * δ_{jk} was set equal to the first number listed where input k was included in the basis function for the emulator in Table 1 and to the second number for k not included (so that δ_{jk} varies with both basis function and input dimension).*

intervals. The fixed coefficient emulator (1.1) was also fitted to the initial design. Prior judgments were the same as those given above for the constant term, except the variances σ_0^2 were set so that the prior variance of the fixed coefficient emulator and varying coefficient emulator were the same for $x_i = 0.5$ for all $i = \{1, \ldots, 14\}$. For the fixed coefficient emulator, 12 (renewables emulator), 10 (emissions emulator) and 15 (cost emulator) of the probability intervals did not contain the true computer model output, suggesting a better fit with the varying coefficient emulator. A more detailed comparison of the varying coefficient emulator and the fixed coefficient emulator is given for the full design of the motivating example in subsection 5.2 and for an illustrative example in section SM1 of the supplementary material.

Figure 1 shows that for a small selection of design points, the variance of the emulator is very large in comparison to the range of each output (e.g. one of the probability intervals for the cost spans over £4bn). Given the small size of the design, large probability intervals are to be expected. The probability intervals in the plots shown could be considered as worst case scenarios, as most of the design points form a space-filling Latin Hypercube sample. Performing a leave-one-out validation on a Latin Hypercube design results in outputs being estimated using the emulator at points with no design points nearby, so the emulator variance is likely to be larger. Ideally, a separate test set would be used to validate the emulator, but the small size of the design prevented this.

In the next section, 24 further design points are selected. By increasing the size of the design the fit of the emulator should improve. The process used to select the design points focuses on reducing the variance of the emulator in regions of the input space of interest rather than using a space-filling design, as done for the initial set of design points. By using a criterion-based approach, the aim is to make best possible use of the limited number of computer model runs available.

4. Design selection. In Subsection 3.1, a varying coefficient model was used to emulate a complex computer model. The aim when fitting an emulator is often to reduce uncertainty about some decision based on a computer model as much as possible, whilst accurately modelling the uncertainty associated with this decision. The varying coefficient model in subsection 3.1 can be used to accurately capture the uncertainty arising from a small design but in order to reduce this uncertainty it is necessary to pair the varying coefficient model

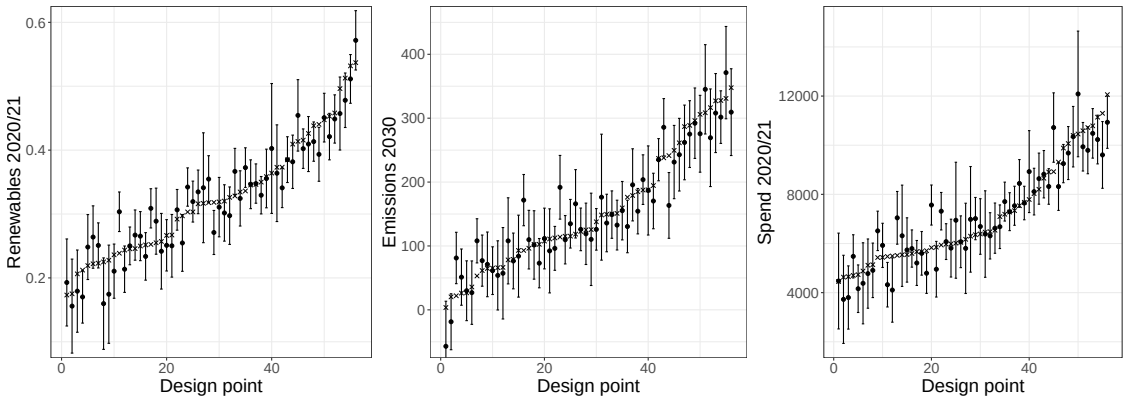

Figure 1. *Leave-one-out plot to assess the fit of the emulator for the proportion of renewable generation in 2020/21 (left), emissions in 2030 (middle) and cost in 2020 (right). Computer model outputs shown with a cross, emulator prediction shown by a circle with probability interval formed by taking two standard deviations either side of emulator prediction. Design points are shown ordered by their respective outputs.*

with a carefully chosen design. This is particularly important when it is not possible to reduce uncertainty by performing further batches of model evaluations. In this section, methodology for sequentially selecting a small design for an emulation study is described and applied to the motivating example described in subsection 2.3.

The steps for choosing a design described here are not specific to the emulator given in (3.1). It is assumed that model evaluations are done in waves, with N_m model evaluations selected as the design for the m-th wave. The design for the m-th wave uses data from waves $1, \ldots, m-1$. The methodology presented is criterion-based, i.e. the aim is to select a design for the next wave, D_{new}, which minimises or maximises some criterion. A good choice of criterion is one which improves the fit of the emulator in the region of space most of interest in the study. For example, an emulator might be used to find the input to a computer model that results in the maximum possible output. A good criterion in this case would focus on reducing uncertainty in the emulator in regions of $\mathbf{x}$ such that $\mathbf{f}(\mathbf{x})$ is likely to be high.

4.1. Criterion. The aim is to select a new design D_{new} which optimises some criterion. To construct this criterion we use a combination of some weights and a utility function. The utility function, denoted $U(\mathbf{x})$, is used to assess the utility of the emulator at some input $\mathbf{x}$ if the new design were chosen (for example $U(\mathbf{x})$ could be the emulator variance at $\mathbf{x}$). The weights $w(\mathbf{x})$, where $0 \leq w(\mathbf{x}) \leq 1$, are used to reflect the relative importance of the input $\mathbf{x}$ in relation to the purpose of the emulation study. These weights can be used to focus the design on improving the emulator in the region of the input space most of interest. The utility function and the weights are combined to form a criterion $C(D_{new})$ for some new design D_{new}:

$$C(D_{new}) = \sum_g w(\mathbf{x}^{(g)}) \mathbb{E}^{f_i(D_{new})} \left[U(\mathbf{x}^{(g)}) \right], \tag{4.1}$$

where $X = \{\mathbf{x}^{(g)} \text{ for } g \in \{1, \ldots N_g\}\}$ is a grid of points over which the criterion is evaluated, $D_{new} = \{x^{(d)} \text{ for } d \in \{1, \ldots N_D\}\}$ is a new design and $f_i(D_{new})$ is the vector of outputs with d-th entry $f_i(\mathbf{x}^{(d)})$. The notation $\mathbb{E}^{f_i(D_{new})}$ in (4.1) is used to mean that the expectation is taken over the distribution of the outputs $f_i(D_{new})$ associated with the new design D_{new}.

Taking the expectation over the distribution of $f_i(D_{new})$ is necessary because $U(\mathbf{x})$ is used to evaluate the new design, and hence may be dependent on the unknown outputs $f_i(D_{new})$.

In section 3 the dimensions of the emulator were assumed to be independent of one another. This simplification is continued here. It is theoretically possible to extend the approach described here to the multivariate case by replacing $f_i(D_{new})$ by the matrix of outputs $\mathbf{f}(D_{new})$ with (i,d)-th entry $f_i(\mathbf{x}^{(d)})$ but in practice this may be challenging for applications with small designs. In the rest of this section, the subscript i on $f_i(D_{new})$ is dropped to simplify the notation.

4.2. Example: criterion selection.

4.2.1. Objectives of study. Limited computing time was available to make further runs of the computer model described in subsection 2.3 and so it was necessary to carefully select which sets of inputs to test. In section 3 the varying coefficient emulator was fitted to an initial Latin hypercube design of size 56. The criterion-based methods described above were used to choose an additional 24 design points. This section describes how the general principles described above for choosing a criterion were applied specifically to the example.

The aim of the analysis was to choose a set of strike prices that were likely to result in the meeting of the three government objectives associated with cost, proportion of renewables and emissions. Whether a set of strike prices will meet the three objectives is uncertain. This uncertainty stems from parametric uncertainty in $\mathbf{z}$, uncertainty as to the output of the computer model for any given input (given that the number of model evaluations is limited) and the structural discrepancy of the computer model in comparison to the real-world. The aim can therefore be thought of as finding a set of strike prices which is associated with a high probability of meeting the three objectives given these uncertainties.

From the perspective of a decision-maker, the three objectives may have unequal weight. For example, it may be preferable to have a higher certainty of meeting the renewable target at the expense of the emissions target, given that the renewables target occurs ten years before the emissions target so there is more time to make policy changes. As a result, multiple solutions may be of interest to decision-makers, depending on the relative importance of each objective to that decision-maker. It is not necessarily the case that the optimal solution is that which minimises the cost subject to the constraints on renewable generation and emissions. As such, rather than focussing on design strategies which would help locate the strike prices with the minimum expected spend subject to the constraints on renewables and emissions, we develop a design criterion that attempts to reduce the variance of the emulator (integrated over the parametric uncertainty) in the region of the input space where the renewable, emissions and cost constraints have a high probability of being met. The benefit of such an approach is the ability to present decision-makers with a range of options with different expected costs, proportions of renewable generation and emissions along with the associated uncertainties. Decision-makers can then use this evidence to determine their own view of the best choice of strike price for each technology.

4.2.2. Choosing the utility function and weights. Recall that $f_r(\theta,\mathbf{z})$, $f_e(\theta,\mathbf{z})$ and $f_s(\theta,\mathbf{z})$ are emulators for the proportion of renewable generation, the CO2 emissions and the total cost of the support scheme. We let our parametric uncertainty for $\mathbf{z}$ be described by the probability density function $p_Z(\mathbf{z})$. As described above, the aim when selecting the new design is to reduce the variance of the emulator (integrated over the parametric

uncertainty) in the region of the input space of interest. The utility function is thus set to

$$U(\theta) = \mathrm{Var}^*(\mathbb{E}^Z[f_r(\theta, \mathbf{z})]) + \mathrm{Var}^*(\mathbb{E}^Z[f_e(\theta, \mathbf{z})]) + \mathrm{Var}^*(\mathbb{E}^Z[f_s(\theta, \mathbf{z})]), \tag{4.2}$$

where Var^* is the variance arising from functional uncertainty (i.e. the variance due to emulation) and $\mathbb{E}^Z$ is the expectation taken over the parametric uncertainty given by the joint probability density function $p_Z(\mathbf{z})$. In the analysis done here a specific choice is made for $p_Z(\mathbf{z})$ (given later), but it would be possible to test a range of distributions. At any given θ, the value of $\mathbb{E}^Z[f(\theta, \mathbf{z})]$ (dropping the subscript r, e or s) is uncertain, because the function f is uncertain. The uncertainty in f is modelled using the emulator. The term $\mathrm{Var}^*(\mathbb{E}^Z[f(\theta, \mathbf{z})])$ measures the variance in $\mathbb{E}^Z[f(\theta, \mathbf{z})]$ arising from uncertainty as to the form of the function f. The utility function was chosen so that new designs are evaluated by the extent to which they reduce this uncertainty.

For this example, the input space has been partitioned into $\mathbf{x} = (\theta, \mathbf{z})$. The strike prices θ are control parameters as they are under the control of policy-makers. The $\mathbf{z}$ are parameters which are uncertain and out of the control of policy-makers (e.g. electricity demand). As the objective is to investigate strike price choices (i.e. choices of θ), we chose to minimise the sum of the variances of $\mathbb{E}^Z[f(\theta, \mathbf{z})]$, rather than of $f(\theta, \mathbf{z})$. This decision was made because summary features at a particular choice of strike price are of more interest to policymakers than the output of the model at a particular value of $\mathbf{z}$. To reflect that $\mathbb{E}^Z[f(\theta, \mathbf{z})]$ is a function of θ, $U(\cdot)$ is a function of θ rather than $\mathbf{x}$.

The computer model inputs and outputs will be scaled before fitting the emulator. As a result, the variance of each dimension of the emulator is on the same scale and so in (4.2), the variances of the three emulators are given equal weight. In cases where there is a greater tolerance for uncertainty in a subset of the emulators, it would be possible to adjust the contribution of each emulator to the overall utility function.

The weights $w(\cdot)$ are set to

$$w(\theta) = P(f_r(\theta, \mathbf{z}) + \epsilon_r > c_r, f_e(\theta, \mathbf{z}) + \epsilon_e < c_e, f_s(\theta, \mathbf{z}) + \epsilon_s < c_s), \tag{4.3}$$

where ϵ_r, ϵ_e and ϵ_s are three independent Normally distributed random variables used to model the discrepancy between the computer model and the real-world and $c_r = 0.3$, $c_e = 100$ and $c_s = 7600$. Further details about this discrepancy are given later in section 5. The values of c_r, c_e and c_s are set to the government targets (of 30% renewable generation, CO2 emissions less than 100gCO2/kWh and a cost of less than £7.6 bn). The probability in (4.3) accounts for parametric uncertainty, structural discrepancy and functional uncertainty.

The function $U(\cdot)$ measures the sum of the variances of the expected values of the three emulators (where the expected value is taken over parametric uncertainty) at a given grid point. This is an appropriate choice because our interest is in the cost, the proportion of renewables and the emissions accounting for parametric uncertainty, so it is desirable to reduce our uncertainty about these quantities. The weights $w(\cdot)$ give more weight to grid points with a higher probability of meeting the three government targets. By combining the weights and the utility function, designs which reduce the emulator variance in the region of the input space of interest will be prioritised over designs which reduce the emulator variance elsewhere.

4.3. Estimating the criterion. Depending on the choice of utility function, it can sometimes be possible to determine analytically the value of $\mathbb{E}^{f(D_{new})}[U(\mathbf{x}^{(g)})]$. For cases in which the utility function depends on the uncertain outputs $f(D_{new})$, simulation of

$f(D_{new})$ to pair with D_{new} can be used to estimate this expectation. The following steps describe a procedure for simulating $f(D_{new})$ and estimating the criterion for a given design $D_{new} = \{\mathbf{x}^{(d)} \text{ for } d \in \{1, \ldots, N_D\}\}$, incorporating model evaluations from all previous waves, D.

1. Draw a grid of size N_g over the input space $\mathbf{x}$ using a space-filling design such as a Latin Hypercube sample. Denote this grid $X = \{\mathbf{x}^{(g)} \text{ for } g \in \{1, \ldots N_g\}\}$.
2. Use the Bayes linear updating equations (3.4) and (3.5) to obtain adjusted beliefs for $f(\mathbf{x})$ using all previously run model evaluations D. From this updated emulator, the adjusted mean $\mathbb{E}_D[f(D_{new})]$ and covariance matrix $\mathrm{Cov}_D[f(D_{new})]$ for $f(D_{new}) = (f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(N_D)}))$ can be obtained . Approximate the joint distribution of $f(D_{new})$ by some probability distribution with mean and covariance given by $\mathbb{E}_D[f(D_{new})]$ and $\mathrm{Cov}_D[f(D_{new})]$ respectively. For the motivating example, we use the multivariate Normal distribution for this approximation. Draw a sample of size N_f from this distribution, and denote the j-th draw from this sample $f(D_{new})^{(j)}$.
3. The criterion can then be estimated by

$$C(D_{new}) \approx \sum_g w(\mathbf{x}^{(g)}) \frac{1}{N_f} \sum_{j=1}^{N_f} U(\mathbf{x}^{(g)} : f(D_{new})^{(j)}), \tag{4.4}$$

   where the notation $U(\mathbf{x}^{(g)} : f(D_{new})^{(j)})$ is used here to indicate that the utility function should be evaluated using the simulated model evaluations $f(D_{new})^{(j)}$ paired with the design D_{new} (as well as all previous model evaluations D).

The procedure described above can be used to estimate the criterion $C(\cdot)$ for a large number of candidate designs. These designs can then be compared and the design with the largest (or smallest) criterion chosen. Alternately, a stepwise addition and/or deletion procedure can be used.

Estimating the criterion by simulation as described above can be a computationally expensive procedure. For each set of simulated outputs $f(D_{new})$ the utility function must be estimated. As the aim is usually to improve the emulator in some way it can be necessary (depending on the chosen utility function) to re-fit the emulator with each simulated $f(D_{new})$. The full procedure must then be repeated for each candidate design D_{new}. If MCMC is needed to re-fit the emulator each time a new $f(D_{new})$ is drawn it is likely to be computationally intractable to estimate the criterion using simulation for a reasonable number of candidate designs. Using Bayes Linear methods rather than a full probability specification make the above computations feasible by reducing fitting the emulator to straightforward linear algebra.

4.4. Example: estimating the criterion. To estimate the criterion for some design D_{new} for the motivating example, the expected value of the utility function (4.2) under this new design must be estimated. To estimate this expectation, simulations of $f(D_{new})$ to pair with D_{new} are needed. The steps given above can be used to draw these simulations. As the utility function requires integration over the uncertain parameters $\mathbf{z}$, the criterion in step 3 above cannot be evaluated analytically. This section describes the simulation procedure used to estimate the utility function and the weights need in step 3 above for the motivating example. The steps taken here are specific to the utility function chosen for the example.

The utility function (4.2) was estimated for each proposed new design D_{new} (and the simulated outputs paired with this new design $f(D_{new})$) using the Monte Carlo simulation steps below. These steps were repeated for each dimension of the emulator, with f replaced by each of f_r, f_e and f_s.

1. Draw $\mathbf{z}^{(i)}$ for $i \in \{1, \ldots, I\}$ from $p_Z(\mathbf{z})$. The distribution $p_Z(\mathbf{z})$ describes uncertainty in the inputs $\mathbf{z}$, the distribution used is given in subsection 5.3.
2. Update the emulator using (3.4) and (3.5), where D in this case will consist of the existing model evaluations and the proposed new design D_{new} (where the outputs paired with this new design have been sampled using the procedure described above). Obtain the mean-vector M of length I with i-th entry $\mathbb{E}_D[(f(\theta, \mathbf{z}^{(i)})]$ and the $I \times I$ covariance matrix C with (i, j)-th entry $\mathrm{Cov}_D[f(\theta, \mathbf{z}^{(i)}), f(\theta, \mathbf{z}^{(j)})]$ from this updated emulator.
3. Draw K emulators by taking K draws from a multivariate Normal distribution with mean M and covariance matrix C. Denote the k-th draw $\mathbf{f}^{(k)}(\theta) = f^{(k)}(\theta, \mathbf{z}^{(1)}), \ldots, f^{(k)}(\theta, \mathbf{z}^{(I)})$.
4. Estimate

$$\mathbb{E}^*[\mathbb{E}^Z[f(\theta, \mathbf{z})]] \approx \frac{1}{K} \sum_{k=1}^{K} \frac{1}{I} \sum_{i=1}^{I} f^{(k)}(\theta, \mathbf{z}^{(i)})$$

$$\mathrm{Var}^*(\mathbb{E}^Z[f(\theta, \mathbf{z})]) \approx \frac{1}{K-1} \sum_{k=1}^{K} \left(\frac{1}{I} \sum_{i=1}^{I} f^{(k)}(\theta, \mathbf{z}^{(i)}) - \mathbb{E}^*[\mathbb{E}^Z[f(\theta, \mathbf{z})]] \right)^2.$$

The estimate of the utility function is then given by the sum of $\mathrm{Var}^*(\mathbb{E}^Z[f_r(\theta, \mathbf{z})])$, $\mathrm{Var}^*(\mathbb{E}^Z[f_e(\theta, \mathbf{z})])$ and $\mathrm{Var}^*(\mathbb{E}^Z[f_s(\theta, \mathbf{z})])$.

The terms $\mathbb{E}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]]$ and $\mathrm{Var}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]]$ for $f = f_r, f_e, f_s$ can also be approximated using the process described above, but with estimators

$$\mathbb{E}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]] \approx \frac{1}{K} \sum_{k=1}^{K} \frac{1}{I-1} \sum_{i=1}^{I} \left(f^{(k)}(\theta, \mathbf{z}^{(i)}) - \frac{1}{I} \sum_{i=1}^{I} f^{(k)}(\theta, \mathbf{z}^{(i)}) \right)^2,$$

$$\mathrm{Var}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]] \approx \frac{1}{K-1} \sum_{k=1}^{K} \left(\frac{1}{I-1} \sum_{i=1}^{I} \left(f^{(k)}(\theta, \mathbf{z}^{(i)}) - \frac{1}{I} \sum_{i=1}^{I} f^{(k)}(\theta, \mathbf{z}^{(i)}) \right)^2 - \mathbb{E}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]] \right)^2.$$

These terms are not used in the utility function but are used later in this paper when comparing summary statistics associated with different choices of strike prices.

To estimate each weight (4.3), the process described above for the estimation of the utility function can be used, but the design points D will consist only of the existing model evaluations. The estimate of $w(\theta)$ obtained from K emulator draws $\mathbf{f}^{(1)}(\theta), \ldots, \mathbf{f}^{(K)}(\theta)$ is

given by

$$w(\theta) \approx \frac{1}{K}\sum_{k=1}^{K}\frac{1}{I}\sum_{i=1}^{I} P(c_r - f_r^{(k)}(\theta, \mathbf{z}^{(i)}) < \epsilon_r) \times P(c_e - f_e^{(k)}(\theta, \mathbf{z}^{(i)}) > \epsilon_e) \times P(c_s - f_s^{(k)}(\theta, \mathbf{z}^{(i)}) > \epsilon_s),$$

where independence holds because the terms ϵ_r, ϵ_e and ϵ_s are assumed independent. For correlated error terms, the joint probability must be evaluated. For the above computations, K was set to 10,000 and I to 2,000. These values were found to give a good balance between computation time and Monte Carlo error when tested.

4.5. Example: design selection for motivating example. The first wave of the analysis was described in subsection 3.3. The 56 design points run in the first wave were used to construct emulators f_r, f_e and f_s as described in subsection 3.3.2. For the second wave of analysis, the methodology described above was used to select a further 16 design points. A maximin Latin hypercube of size 50, $X = \{x_1, \ldots, x_{50}\}$, was generated over all fourteen inputs. Design points were chosen from this Latin hypercube sample using the following procedure:

1. Set $D_{new} = \{x_i\}$ for each x_i in the Latin hypercube sample and estimate the criterion (4.1) for D_{new}. Repeat for $i \in \{1, \ldots, 50\}$.
2. Order the points in the Latin hypercube sample by the size of the criterion, denoting this ordered sample $x^{(1)}, x^{(2)}, \ldots, x^{(50)}$. Set $D_{new} = \{x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}\}$, selecting the four points with the smallest criterion.
3. Set $D_{new} = \{x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}, x^{(i)}\}$ and estimate the criterion for D_{new} for $i \in \{5, \ldots, 50\}$. Add the four points associated with the smallest criterion to the design.
4. Repeat the above process for another step, to add four further points to the design, giving a design of size 12, which we denote D_x.
5. Draw a new Latin hypercube sample $X_2 = \{x_{2,1}, \ldots, x_{2,50}\}$ of size 50. Set $D_{new} = \{D_x, x_{2,i}\}$ and estimate the criterion for D_{new} for $i \in \{1, \ldots, 50\}$. Add the four points from X_2 associated with the smallest criterion to D_{new}.

The choice to restrict the Latin hypercube to 50 design points and to choose additions to the design in batches of four (rather than using stepwise addition or deletion) was made to reduce the computation time. Further technical details on design selection specific to the motivating example including a summary of the design points selected can be found in section SM3 of the supplementary material.

Plots showing the criterion evaluated for each candidate design point at each stage of the design selection process are shown in Figure 2. The plot on the left shows the criterion for each of the 50 points in the initial set of candidate points. The four points with the lowest criterion were selected. In the next plot the criterion for the remaining 46 candidate design points is shown, having incorporated the four points selected in the previous stage into the design. In the plot on the right a new set of 50 candidate design points were tested. At each subsequent stage, the criterion reduces as more points are added to the design and the size of this improvement decreases with each stage. It is clear by comparing the first plot to the third plot that as more design points are chosen from the initial Latin hypercube sample the difference in criterion between the best possible point and the worst possible point decreases.

The criterion was evaluated incorporating all sixteen of the additional design points into D_{new} and was found to be 0.0226 (or 0.0227 for the actual reduction in variance

rather than expected), compared to 0.0274 for a Latin hypercube sample. These values correspond approximately to a reduction in standard deviation of £45m for cost, 0.2% for proportion of renewable generation and 2gCO2/kWh of emissions for a given set of strike prices (dividing the reduction in variance evenly between the three emulators).

In practice, it was possible to run a further 8 model evaluations in addition to the 16 selected above. The five points with the highest prior probability of meeting the objectives (i.e. with the highest weights) out of a possible 1,000 points tested were run. Three further design points were then chosen from the second design X_2.

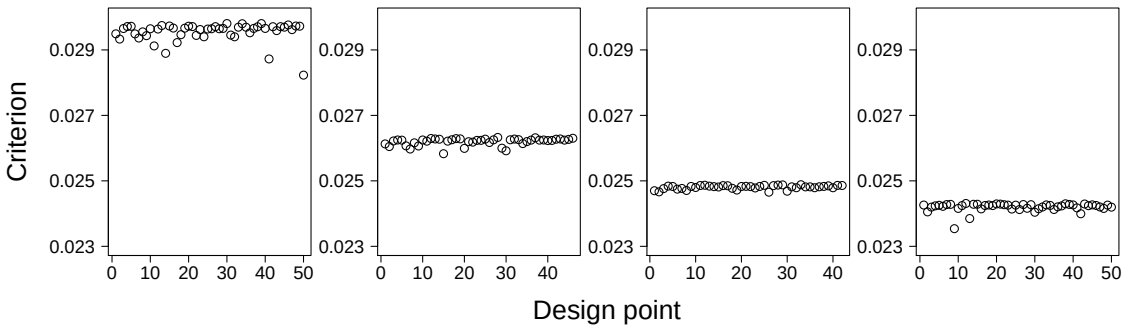

Figure 2. *Estimated criterion for candidate design points. From left to right, corresponding to steps 2, 3, 4, 5 in the text. All criterion estimated for emissions of* $100gCO2/kWh$.

5. Example: results. The procedure described in section 4 was used to select 24 further model evaluations for the electricity supply model, in addition to the 56 discussed in subsection 3.3. These 24 model evaluation were run and an emulator fitted to the full set of 80 model evaluations. This section describes the emulator that was fitted and presents results from the study.

As for the initial design, the varying coefficient emulator (3.1) was used to model the three computer model outputs: renewable generation in 2020, emissions in 2030 and spend in 2020. Each of these outputs was assumed to be independent conditional on the input parameters. To incorporate structural discrepancy, the real-world output $\mathbf{y}(\theta, \mathbf{z}) = (y_r(\theta, \mathbf{z}), y_e(\theta, \mathbf{z}), y_s(\theta, \mathbf{z}))$ (with subscript r for renewables, e for emissions and s for spend) was modelled as

$$\mathbf{y}(\theta, \mathbf{z}) = \mathbf{f}(\theta, \mathbf{z}) + \epsilon, \tag{5.1}$$

where $\mathbf{f}(\theta, \mathbf{z}) = (f_r(\theta, \mathbf{z}), f_e(\theta, \mathbf{z}), f_s(\theta, \mathbf{z}))$ is the emulator for the computer model and $\epsilon = (\epsilon_r, \epsilon_e, \epsilon_s)$ is a vector of independent error terms, with $\epsilon = \text{MVN}(\mathbf{0}, \Sigma)$, for some diagonal covariance matrix Σ (numerical values are given later). The terms ϵ and $\mathbf{f}(\theta, \mathbf{z})$ are assumed to be independent of one another.

5.1. Emulation using full design. The varying coefficient emulator $\mathbf{f}(\theta, \mathbf{z})$ in (5.1) was fitted using the full set of design points tested, i.e. the 80 design points comprising the 56 from the initial design described in subsection 3.3.2 and the 24 from the design described in section 4. The Bayes linear approach described in subsection 3.2 was used to fit the emulator.

As for the emulator based on the initial design (described in subsection 3.3.2), linear regression fits, the residual sum of squares of the emulator and expert judgment were

used to select the basis functions for each output. Prior judgments were adjusted so that the variance term $\sum h_j^2(\mathbf{x})\sigma_j^2$ was approximately equal to the residual variance of a linear regression fit when $x_k = 0.5$ for all $k \in \{1, \ldots, 14\}$. These residual variances were 0.13^2 (renewables model), 0.13^2 (emissions model) and 0.15^2 (spend model). The remainder of the prior assumptions were set as in subsection 3.3.2. Unlike in subsection 3.3.2, values for δ_{jk} were set to the same value for all k. Although it was expected that inputs included in the mean function would have a bigger effect on the correlation (and hence be associated with a smaller δ_{jk}), this was not found to make a difference in practice. Tables listing the basis functions used and numerical values for the prior judgments are given in section SM4 of the supplementary material.

To validate the emulator, leave-one-out plots equivalent to those in Figure 1 were produced to check emulator predictions against the true computer model output. For all three outputs the emulator had a good predictive ability.

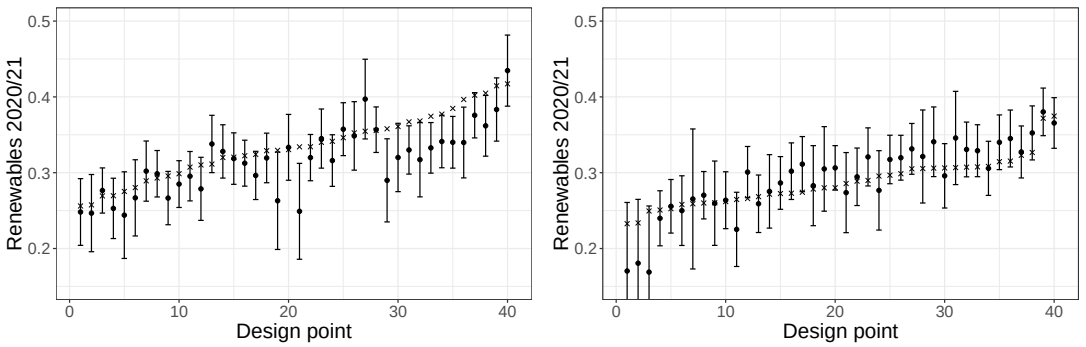

Figure 3. *Assessment of the fit of the fixed coefficient emulator to renewable generation model output. The dataset used to fit the emulator was split into two, based on the value of the offshore load factor. The graph on the LHS was fitted using design points with the scaled offshore load factor less than zero and tested against design points with load factor greater than zero. The graph on the RHS was fitted using those design points with scaled offshore load factor greater than zero and assessed against those with load factor less than zero. Computer model outputs are shown with a cross, emulator prediction shown by a circle with probability interval.*

5.2. Comparison to fixed coefficient emulator. To check whether a varying coefficient emulator was necessary, the fixed coefficient emulator in (1.1) was fitted to different regions of the input space and the predictions using this emulator were tested against the actual computer model output. Figure 3 shows the fixed coefficient emulator predictions for renewable generation when the input space is partitioned by offshore load factor. As can be seen, the emulator seems to be underestimating the proportion of renewable generation in the plot on the left hand side, and overestimating the proportion of renewable generation in the plot on the right hand side. This difference is evidence to suggest that the coefficients B are varying as the load factor for offshore wind varies, and so a varying coefficient emulator is needed. If the coefficients were constant throughout the space then the emulator fitted to design points with load factor less than zero should also be able to predict the output for design points with load factor greater than zero. Coefficient estimates for the fixed coefficient emulator and the varying coefficient emulator were examined for each of the outputs and were found to vary in different regions of the space, further supporting the need for a varying coefficient emulator.

The fit of the varying coefficient emulator was compared to the fit of a fixed coefficient

emulator using the prediction residual sum of squares (Table 3), calculated as the sum of the squared difference between the leave-one-out prediction and the true output. The prior variance term σ^2 of the fixed coefficient emulator for each output was set to the residual variance of a linear regression fit to that output. This is comparable to the prior variance of the varying coefficient emulator at $x_i = 0.5$ for all $i \in \{1, \ldots, 14\}$. For the emulators for emissions and cost, the residual sum of squares is smaller for the varying coefficient emulator, implying that the fit of the varying coefficient model is better. For the emulator for the proportion of renewable generation, the residual sum of squares is the same for both emulators, so there may be no advantage in using a varying coefficient emulator in this case (although it is not clear whether there would be an advantage or not with further extrapolation outside the existing design given the plots in Figure 3). Note that as Table 3 gives the prediction error we do not need to adjust these numbers to account for the additional degrees of freedom required for fitting the varying coefficient emulator.

Output	RSS - non-varying coefficients	RSS - varying coefficients
Renewable generation	1.90	1.89
Emissions	1.79	1.62
Spend	2.50	2.01

Table 3

Residual sum of squares (RSS) from leave-one-out cross validation estimated for each rescaled output for the varying coefficient model compared to the fixed coefficient emulator

The varying coefficient emulator is a more general form of the fixed coefficient emulator, allowing for a more flexible prior specification. In the absence of evidence to suggest that the relationship between the inputs and outputs of a computer model are constant throughout the space it is risky to make this assumption, as the resulting uncertainty in the computer model outputs will be underestimated. As demonstrated above, there is evidence for the motivating example that coefficients are varying and that a varying coefficient emulator gives a better fit than one with fixed coefficients (as measured using the RSS).

5.3. Parametric and structural uncertainty. The aim of the analysis was to find a set of strike prices that will meet the three government targets with a high probability, accounting for uncertainty in the eight input parameters $\mathbf{z}$. To model this parametric uncertainty $\mathbf{z}$ was assumed to have a multivariate Normal distribution with

$$\begin{aligned}
\mathbb{E}[\mathbf{z}] =& (0,0,0,1,1,1,1,1)^T \\
\mathrm{Var}[\mathbf{z}] =& (0.5^2, 0.5^2, 0.5^2, 0.05^2, 0.05^2, 0.05^2, 0.05^2, 0.05^2)^T \\
\mathrm{Corr}(z_1, z_2) =& \mathrm{Corr}(z_1, z_3) = 0.2,\ \mathrm{Corr}(z_2, z_3) = 0.4, \\
\mathrm{Corr}(z_i, z_j) =& 0 \text{ for other pairs of inputs}
\end{aligned}$$

where the dimensions of $\mathbf{z}$ are defined in subsection 3.3.1. The mean of this distribution is the central government scenario used in [27]. The variance of the demand, coal, gas and technology cost parameters was set so that a 95% probability interval around the mean corresponds approximately to the low and high scenarios tested in [27]. The variance for the hurdle rate and load factor parameters was set so that a 95% probability interval around the mean would be approximately $(0.9, 1.1)$ (uncertainty in these parameters was not considered in [27]). The correlation between gas and demand and coal and demand was set to 0.2 to represent that over the short term an increase in electricity demand could

lead to an increase in gas and coal prices as more fuel is needed to meet demand. The correlation between coal price and gas price was set to 0.4 because a rise in the price of one fuel could lead to increased usage of the other fuel, resulting in increased prices in both.

With no historical data available, and no computer model evaluation time to investigate internal discrepancy, the covariance matrix Σ of the error term $\epsilon = (\epsilon_r, \epsilon_e, \epsilon_s)$ was set to a diagonal matrix with diagonal entries $7.7 \times 10^{-6}, 25, 3859$. For renewable generation and spend in 2020, these values represent a standard deviation of around 1% of the target. For emissions in 2030, the standard deviation is 5% of the target. The standard deviations of the errors associated with renewable generation and spend in 2020 are proportionately much smaller than the standard deviation associated with emissions in 2030, to reflect the greater potential for increased structural discrepancy over a longer time period. If a complete analysis were performed for policy purposes, it would be necessary to test the sensitivity of the results to the assumptions for $p_Z(\mathbf{z})$ and ϵ given above. A study using the standard emulation approach to assess sensitivity and uncertainty of wholesale electricity prices to computer model input assumptions is done in [40].

Note that the distributions representing parametric and structural uncertainty were chosen to demonstrate the statistical methodology that is applicable for this problem and are not a quantitatively accurate representation of the GB electricity system. The results in this paper should therefore be seen as illustrative of the methodology and not as assessments of probabilities of meeting government targets.

5.4. Choosing strike prices. The aim of the analysis was to use emulation to investigate possible strike prices for onshore wind, offshore wind and solar generation. The strike prices chosen should be associated with a high probability of meeting government targets for renewable generation, emissions and spend given uncertainty in eight other computer model inputs and structural discrepancy.

Uncertainties were assessed using simulation in combination with the emulator (fitted to all 80 design points). Following previous notation, we denote the expectation and variance taken with respect to the uncertainty modelled using the emulator by $\mathbb{E}^*$ and Var^*. Steps for the estimation of $\mathbb{E}^*[\mathbb{E}^Z[f(\theta, \mathbf{z})]]$, $\mathrm{Var}^*[\mathbb{E}^Z[f(\theta, \mathbf{z})]]$, $\mathbb{E}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]]$ and $\mathrm{Var}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]]$ for $f = f_r, f_e, f_s$ and $P(y_r(\theta, \mathbf{z}) > c_r, y_e(\theta, \mathbf{z}) < c_e, y_s(\theta, \mathbf{z}) < c_s)$ are described in section 4. To obtain the expectations and variances of $y(\theta, \mathbf{z})$ rather than $f(\theta, \mathbf{z})$, the only adjustment required to the quantities above is $\mathbb{E}^*[\mathrm{Var}^Z[y(\theta, \mathbf{z})]] = \mathbb{E}^*[\mathrm{Var}^Z[f(\theta, \mathbf{z})]] + \mathrm{Var}[\epsilon]$.

Prob of meeting objectives				Auction parameter values					
Joint Prob	Renewable target	Emission target	Spend target	Rate of decay offshore	Starting price offshore	Rate of decay solar	Starting price solar	Rate of decay onshore	Starting price onshore
0.14	0.80	0.29	0.81	0.12	142.79	3.14	110.55	0.54	75.56
0.14	0.86	0.46	0.57	0.19	149.58	0.41	80.62	0.38	75.71
0.13	0.87	0.44	0.60	0.03	147.54	1.18	123.17	1.09	76.83
0.13	0.87	0.46	0.56	0.25	150.53	1.44	124.91	2.59	76.79
0.12	0.88	0.52	0.47	0.36	153.90	3.90	92.04	1.37	76.75

Table 4

Joint probability and marginal probabilities of meeting the three targets for the five strike price choices with highest joint probability of meeting government objectives out of 1,000 tested.

A Latin hypercube sample of size 1,000 over the six strike price inputs was generated and the joint and marginal probabilities of meeting the three government targets at each

set of strike prices in this Latin hypercube sample were estimated. The five sets of strike prices with highest joint probability are presented in Table 4. The highest joint probability over all 1,000 sets of points tested is 14% but as Table 4 shows, sets of points with similar joint probabilities can have very different probabilities of meeting the individual targets.

The results from all 1,000 samples are shown in the heat maps in Figure 4. Results are displayed for the three strike price inputs that had the biggest effect on the probabilities of meeting the targets (starting prices for offshore and onshore wind and rate of decay of offshore wind price). As shown in both Table 4 and Figure 4, a high starting price for offshore wind in combination with a low rate of decay over time of this price is needed to meet all three targets. When the starting price of offshore wind is high, there is a slight increase in the probabilities of meeting the spend and emissions targets if the starting price of onshore wind is low. The starting price for offshore wind was modelled with a varying coefficient in the final fitted emulator. The heat maps in Figure 4 show some evidence of complex spatial variability in the relationship between the probabilities of meeting the spend and emissions targets and this starting price in that the starting price only seems to have an effect on the probabilities in a restricted region of the space. However, it is difficult to assess from Figure 4 alone whether there is complex spatial variability in the relationship between the starting price and the cost or the emissions targets directly as the plots in Figure 4 integrate over the uncertain input parameters $\mathbf{z}$. Regardless of whether the plots show complex spatial variability or not, one advantage of the varying coefficient emulator is that it captures any uncertainty that arises because it is not known whether the spatial variability is simple or not where there are no data.

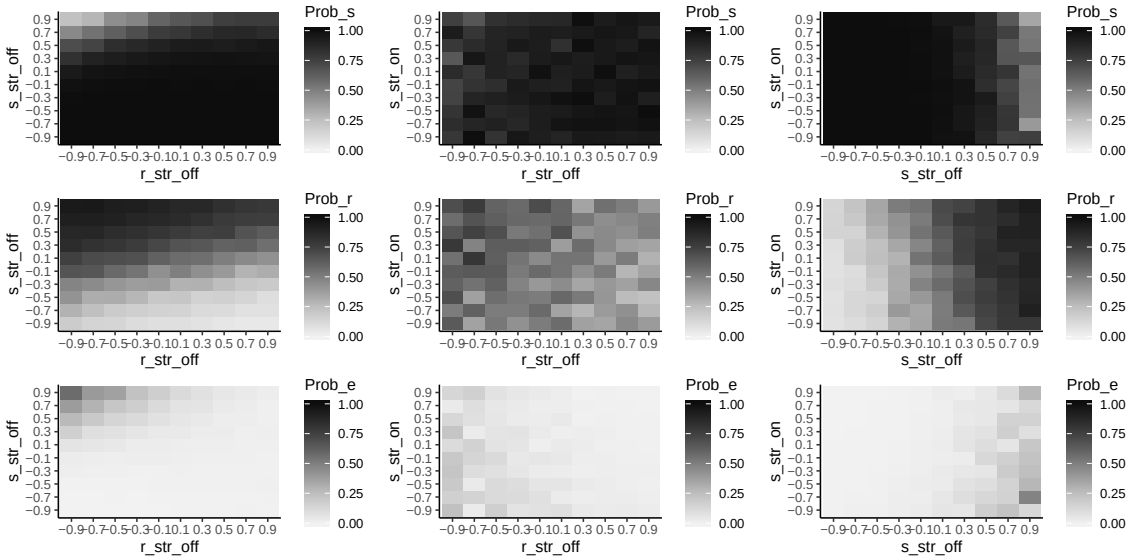

Figure 4. *Heat maps showing the probabilities of meeting the spend (Prob_s), renewable generation (Prob_r) and emissions (Prob_e) targets as the starting strike price for offshore wind (s_str_off), the rate of decay of the offshore wind strike price (r_str_off) and the starting price for onshore wind (s_str_on) vary.*

The strike prices in the top row of Table 4 are explored further in Table 5. The expected proportion of renewable generation in 2020, emissions in 2030 and spend in 2020 given the strike prices in the top row of Table 4 are listed. Standard deviations in these quantities due to parametric uncertainty alone (left hand side) and due to a combination of parametric and

	Parametric		Parametric and structural	
	Mean (SD)	SD (SD)	Mean (SD)	SD (SD)
% Renewables	0.32 (0.0074)	0.025 (0.0023)	0.32 (0.0074)	0.025 (0.0023)
Emissions (gCO2/kWh)	113.1 (3.8)	23.9 (2.2)	113.1 (3.8)	24.4 (2.1)
Cost (£m)	7,138 (85)	532 (39)	7,138 (85)	535 (38)

Table 5

Output uncertainty for strike price choice with highest joint probability of meeting objectives. The left hand side of the table gives the mean and standard deviation with only parametric and functional uncertainty included. The right hand side of the table also incorporates structural discrepancy.

structural uncertainty (right hand side) are given. The effect of the additional structural uncertainty is small in comparison to the parametric uncertainty. The values in brackets in Table 5 give the standard deviations of each of the numbers presented, where this standard deviation arises due to the use of an emulator. The standard deviations due to emulation are still quite large (although small relative to the standard deviation due to parametric uncertainty) but this reflects the small size of the design. These standard deviations could be reduced with further model runs. Whilst the emulator standard deviations seen in Table 5 may be too high for decision-makers to feel confident in making decisions based on these results alone, the results are still of value because they can be used to guide further computer model runs by narrowing down the input space of interest to decision-makers. The size of these errors also highlights the uncertainty that arises when a limited set of model runs is used for analysis. To make fully informed decisions based on computer model output it is necessary to know the extent of this uncertainty.

The results presented in this section demonstrate the importance of presenting to decision-makers a range of possible options, rather than just a single optimal solution (which in this application would be the set of strike prices with the highest joint probability of meeting the targets). With a more complete picture of the possible choices, decision-makers can make a more informed choice. Given the length of time taken for one run of the computer model it would be computationally impossible to investigate the impact of different strike price choices whilst integrating over parametric uncertainty in this way without use of emulation.

6. Discussion. A varying coefficient emulator and design selection procedure were demonstrated on a real-world example involving the selection of parameters in a government policy designed to incentivise investment in renewable technologies to meet government targets.

Motivated by this example, this paper describes methodology for quantifying uncertainty in complex computer models, focussing on situations where the number of available computer model evaluations is small. A varying coefficient model is proposed for emulation of computer model output. For complex models, an assumption that the coefficients of the global mean function of an emulator are constant throughout the input space is unrealistic. When model evaluations are plentiful, the stochastic process governing the local variation can adapt to incorporate these varying coefficients. With limited model evaluations better results may be obtained by incorporating prior knowledge of the varying coefficients into the global mean function. A varying coefficient model also allows for a more accurate representation of uncertainty when extrapolating away from design points, which is of particular concern when the design is small relative to the number of inputs. This is because a varying coefficient model acknowledges uncertainty in the global response surface, as well

as in the local variation.

A design selection procedure for small designs is also presented. When the number of model evaluations is limited, it is crucial to extract maximum benefit from the model evaluations performed. This paper selects a design by estimating the value of some criterion given different possible designs. This criterion can be set as the sum of a utility function, evaluated over a grid, with the contribution of the utility function at each gridpoint determined by a set of weights. The utility function can be used to assess the emulator under the proposed design, and the weights can be used to prioritise the regions of the input space of most interest.

Supplementary material. The supplementary material contains a series of illustrative examples comparing the varying coefficient emulator to a fixed coefficient emulator as well as further details on the inputs of the computer model, the design selection and the fitting of the emulator to the full design.

REFERENCES

[1] S. Ba and V. Joseph, *Composite Gaussian Process models for emulating expensive functions*, Annals of Applied Statistics, 6 (2012), pp. 1838–1860.

[2] K. Chaloner and I. Verdinelli, *Bayesian experimental design: a review*, Statistical Science, 10 (1995), pp. 273–304.

[3] S. Conti, J. P. Gosling, J. E. Oakley, and A. O'Hagan, *Gaussian process emulation of dynamic computer codes*, Biometrika, 3 (2009), pp. 663–676.

[4] S. Conti and A. O'Hagan, *Bayesian emulation of complex multi-output and dynamic computer models*, Journal of Statistical Planning and Inference, 140 (2010), pp. 640 – 651.

[5] P. Craig, M. Goldstein, A. Seheult, and J. Smith, *Bayes linear strategies for matching hydrocarbon reservoir history*, in Bayesian Statistics 5, J. Bernardo, J. Berger, A. Dawid, and A. Smith, eds., Clarendon Press, Oxford, 1996, pp. 69–95.

[6] P. Craig, M. Goldstein, A. Seheult, and J. Smith, *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: Volume 3, C. Gatsonis, J. S. Hodges, R. E. Kass, R. McCulloch, P. Rossi, and N. D. Singpurwalla, eds., Springer, 1997, pp. 37–93.

[7] J. A. Cumming and M. Goldstein, *Small sample bayesian designs for complex high-dimensional models based on information gained using fast approximations*, Technometrics, 51 (2009), pp. 377–388.

[8] J. A. Cumming and M. Goldstein, *Uncertainty analysis for oil reservoirs*, in The Oxford Handbook of Applied Bayesian Analysis, A. O'Hagan and M. West, eds., Oxford University Press, 2010, pp. 241–270.

[9] DECC, *Electricity Market Reform: policy overview (department of energy and climate change)*. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/48371/5349-electricity-market-reform-policy-overview.pdf, 2012. Accessed: 2018-03-12.

[10] DECC, *Electricity Market Reform - contracts for difference (department of energy and climate change)*. https://www.gov.uk/government/collections/electricity-market-reform-contracts-for-difference, 2015. Accessed: 2018-03-12.

[11] A. E. Gelfand, S. Banerjee, C. F. Sirmans, Y. Tu, and S. E. Ong, *Multilevel modeling using spatial processes: Application to the singapore housing market*, Computational Statistics and Data Analysis, 51 (2007), pp. 3567 – 3579.

[12] A. E. Gelfand, H.-J. Kim, C. F. Sirmans, and S. Banerjee, *Spatial modeling with spatially varying coefficient processes*, Journal of the American Statistical Association, 98 (2003), pp. 387–396.

[13] M. Goldstein, *Bayesian analysis of regression problems*, Biometrika, 63 (1976), pp. 51–58.

[14] M. Goldstein, *The linear Bayes regression estimator under weak prior assumptions*, Biometrika, 67 (1980), pp. 621–628.

[15] M. Goldstein and D. Wooff, *Bayes linear statistics, theory and methods*, vol. 716, John Wiley & Sons, 2007.

[16] R. B. Gramacy and H. K. H. Lee, *Bayesian treed gaussian process models with an application to computer modeling*, Journal of the American Statistical Association, 103 (2008), pp. 1119–1130.
[17] R. B. Gramacy and H. K. H. Lee, *Adaptive design and analysis of supercomputer experiments*, Technometrics, 51 (2009), pp. 130–145.
[18] N. A. S. Hamm, A. O. Finley, M. Schaap, and A. Stein, *A spatially varying coefficient model for mapping PM10 air quality at the european scale*, Atmospheric Environment, 102 (2015), pp. 393 – 405.
[19] D. Higdon, J. Gattiker, B. Williams, and M. Rightley, *Computer model calibration using high-dimensional output*, Journal of the American Statistical Association, 103 (2008), pp. 570–583.
[20] M. C. Kennedy and A. O'Hagan, *Bayesian calibration of computer models*, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63 (2001), pp. 425–464.
[21] H.-M. Kim, K. B. Mallick, and C. C. Holmes, *Analyzing nonstationary spatial data using piecewise gaussian processes*, Journal of the American Statistical Association, 100 (2005), pp. 653–668.
[22] L. A. Lee, K. J. Pringle, C. L. Reddington, G. W. Mann, P. Stier, D. V. Spracklen, J. R. Pierce, and K. S. Carslaw, *The magnitude and causes of uncertainty in global model simulations of cloud condensation nuclei*, Atmospheric Chemistry and Physics, 13 (2013), pp. 8879–8914.
[23] J. L. Loeppky, J. Sacks, and W. J. Welch, *Choosing the sample size of a computer experiment: A practical guide*, Technometrics, 51 (2009), pp. 366–376.
[24] M. D. Mckay, R. J. Beckman, and W. J. Conover, *A comparison of three methods for selecting values of input variables in the analysis of output from a computer code*, Technometrics, 21 (1979), pp. 239–245.
[25] M. D. Morris and T. J. Mitchell, *Exploratory designs for computational experiments*, Journal of Statistical Planning and Inference, 43 (1995), pp. 381 – 402.
[26] National Audit Office, *The Levy Control Framework*. https://www.nao.org.uk/report/levy-control-framework-2/, 2013. Accessed: 2018-03-12.
[27] National Grid, *National Grid Electricity Market Reform analytical report*. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/267614/Annex_D_-_National_Grid_EMR_Report.pdf, 2013. Accessed: 2018-03-12.
[28] D. Nychka, D. Hammerling, M. Krock, and A. Wiens, *Modeling and emulation of nonstationary Gaussian fields*, Spatial Statistics, 28 (2018), pp. 21–38.
[29] J. E. Oakley and A. O'Hagan, *Bayesian inference for the uncertainty distribution of computer model outputs*, Biometrika, 89 (2002), pp. 769–784.
[30] J. E. Oakley and A. O'Hagan, *Probabilistic sensitivity analysis of complex models: a Bayesian approach*, Journal of the Royal Statistical Society: Series B, 66 (2004), pp. 751–769.
[31] A. O'Hagan, *Curve fitting and optimal design for prediction*, Journal of the Royal Statistical Society. Series B (Methodological), 40 (1978), pp. 1–42.
[32] C. Pope, J. Gosling, S. Barber, J. Johnson, T. Yamaguchi, G. Feingold, and P. Blackwell, *Gaussian Process modeling of heterogeneity and discontinuities using Voronoi tessellations*, Technometrics, 63 (2021), pp. 53–63.
[33] J. Rougier, *Efficient emulators for multivariate deterministic functions*, Journal of Computational and Graphical Statistics, 17 (2008), pp. 827–843.
[34] T. J. Santner, B. J. Williams, and W. I. Notz, *The design and analysis of computer experiments*, Springer Science & Business Media, 2003.
[35] G. Stephenson, *Review of non-stationary methods for GP emulation*, (2011).
[36] N. M. Urban and T. E. Fricker, *A comparison of Latin hypercube and grid ensemble designs for the multivariate emulation of an earth system model*, Computers and Geosciences, 36 (2010), pp. 746 – 755.
[37] I. Vernon, M. Goldstein, and R. G. Bower, *Galaxy formation: a Bayesian uncertainty analysis*, Bayesian Analysis, 5 (2010), pp. 619–670.
[38] I. Vernon, J. Liu, M. Goldstein, J. Rowe, J. Topping, and K. Lindsey, *Bayesian uncertainty analysis for complex systems biology models: emulation, global parameter searches and evaluation of gene functions*, BMC Systems Biology, 12 (2018).
[39] V. Volodina and D. Williamson, *Diagnostics-driven nonstationary emulators using kernel mixtures*, SIAM/ASA Journal of Uncertainty Quantification, 8 (2020), pp. 1–26.
[40] A. L. Wilson, C. J. Dent, and M. Goldstein, *Quantifying uncertainty in wholesale electricity price projections using Bayesian emulation of a generation investment model*, Sustainable Energy, Grids and Networks, 13 (2018), pp. 42–55.