Weighted space-filling designs

VE Bowman[1] and DC Woods[2*]

[1]*Defence Science and Technology Laboratory, Porton Down, UK;*
[2]*University of Southampton, Southampton, UK.*

Many computer models or simulators have probabilistic dependencies between their input variables, which if not accounted for during design selection may result in a large numbers of simulator runs being required for analysis. We propose a method which incorporates known dependencies between input variables into design selection for simulators and demonstrate the benefits of this approach via a simulator for atmospheric dispersion. We quantify the benefit of the new techniques over standard space-filling and Monte Carlo simulation. The proposed methods are adaptations of computer-generated spread and coverage space-filling designs, with "distance" between two input points redefined to include a weight function. This weight function reflects any known multivariate dependencies between input variables and prior information on the design region. The methods can include quantitative and qualitative variables, and different types of prior information. Novel graphical methods, adapted from fraction of design space plots, are used to assess and compare the designs.

Keywords: computer experiments; defence studies; design of experiments; simulation experiments

1. Introduction

The simulation of physical and engineering systems via complex mathematical models has become a common method of gaining knowledge about mechanisms where physical experimentation would be time consuming, costly or hazardous. In order to explore and understand these often computationally expensive computer codes, experiments are performed where the so-called statistical treatments are combinations of values of the input variables and the responses are deterministic outputs from the computer code, or simulator. See Santner et al. (2003) and Fang et al. (2006) for book-length reviews of the statistical approaches to this topic.

Our motivation for studying these simulators is the application of atmospheric dispersion models (see Section 1.1) used to model deliberate or accidental chemical and biological releases; for example, from acts of terrorism or industrial accidents. Clearly, it is not possible to regularly run highly instrumented field trials to explore the behaviour of a hazardous gaseous release in the atmosphere, and hence computer simulators are important planning and operational tools.

In this paper, we assume that the output from the simulator is deterministic; this is often the case in many engineering and physical science applications (Kennedy et al., 2005). We treat our simulator as an unknown function mapping the simulator inputs $\boldsymbol{x} = (x_1, \ldots, x_k)^{\mathrm{T}}$ to the simulator output y, i.e. $y = f(\boldsymbol{x}; \boldsymbol{\theta})$. The tuning parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^{\mathrm{T}}$ may be treated as additional simulator inputs, calibrated using real physical data (see, for example, Kennedy and O'Hagan, 2001 and Overstall and Woods, 2013) or, as in the dispersion example, determined by subject expert scientists to match observed scenarios. Throughout the rest of this paper, we shall assume that $\boldsymbol{\theta}$ is fixed and
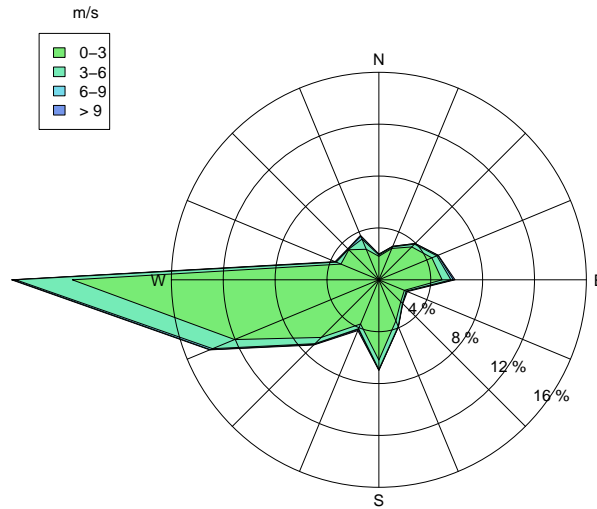

Figure 1: Example of prior information - wind rose for wind speed and direction.

known. This definition of a simulator, although common in the Statistics and Uncertainty Quantification literature, is somewhat different from the stochastic simulation models typically employed in operational research applications; for example, discrete-event or agent-based models (Allen, 2011).

A successfully calibrated simulator can be used to gain scientific understanding of the system and to aid decision-making. Typically, there is uncertainty in the inputs $\boldsymbol{x}$ to the simulator; for example, a dispersion simulation may need to incorporate uncertain meteorology (see Section 1.1). Thus it is necessary to run these simulators multiple times for a variety of combinations of input values in order to understand and quantify the resulting uncertainties in the output (De Rocquigny et al., 2008).

A key aspect of these deterministic simulators is that multiple simulator evaluations at the same combination of input values produce the same response. Hence, repetition of combinations of simulator inputs is not beneficial and the designs commonly used are chosen to cover, or space-fill, the design region of possible input values (Fang, 1980). However, many of these space-filling designs, such as computer-generated designs based on Euclidean distance, do not take into account probabilistic dependencies between input variables. This can result in the selection of points in areas of the design region of little relevance, for example, where it is known that no response can occur. For the dispersion simulator, the meteorological inputs at a particular location are defined, among other things, by wind speed and direction using a wind rose (Figure 1), and can be highly dependent. For example, high wind speeds can be particularly unlikely for certain wind directions and thus simulator runs with these input combinations will reveal little about the behaviour of dispersion at the given location.
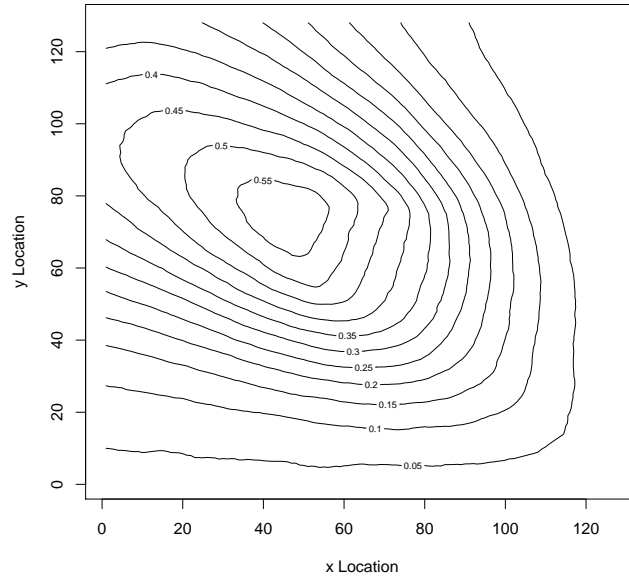
Figure 2: The surface obtained from averaging the number of plumes seen at each coordinate in a plane from 80,000 generated plumes.

1.1. Motivating example

Understanding and predicting how chemical, biological, radiological or nuclear (CBRN) releases will disperse across a given terrain is an important task facing military and civilian emergency planners. Such releases and their subsequent dispersion are therefore simulated using computer codes. Available simulators range from quite simple Gaussian plume models (e.g. Clarke, 1979), through Gaussian puff models (e.g. Sykes et al., 1998) to Lagrangian models (e.g. NAME; Jones et al., 2007). These simulations are undertaken for a number of purposes. In this paper we consider their use for optimizing the placement of CBRN sensors to protect against non-visible airborne pollutants, such as chemicals, particulates and biological agents. For some pollutant sensors, the benefit gained is dependent on where the sensors are placed. Several factors must be considered when optimizing sensor placement, including meteorological conditions and geographical features.

Atmospheric dispersion models typically have the following features:

 (i) the input variables are usually of two types, describing either the meteorology or the source of the release, and can be quantitative or qualitative;

 (ii) there is substantial prior information about the distribution of the input variables from, for example, physical observations (meteorological) or expert prior knowledge (source);

(iii) these distributions are not usually independent, either within type (for example, wind direction and speed are defined via a wind rose) or between type (wind direction and source location);

3

(iv) the distributions define a joint probability density (or weight function) on the design region, which is likely to have substantial areas of low weight.

The sensor placement tool developed by Dstl uses a moderately intensive Gaussian puff dispersion model to simulate releases. For each run of the simulator, our data are a *plume*, defined to be the integral of all puffs from a source over time. The plumes are generated by Monte-Carlo sampling over input variables, to create a response surface on a 128×128 grid, which is the average number of plumes seen at each location. Upwards of 80,000 simulated plumes are required to form a consistent mean surface, resulting in extremely long runtimes; smaller sample sizes cause variability between samples and inconsistent placement of sensors.

Figure 2 shows the output from a Monte Carlo study using 80,000 simulator runs. The effect of wind direction on the response is clear, with a dominant wind blowing from the east pushing the majority of the plumes to the west.

1.2. Aim of the work

In the literature, a variety of approaches have been taken to finding constrained space-filling designs under deterministic constraints, see Iman and Conover (1982), Stinstra et al. (2003), Petelet et al. (2010) and Draguljić et al. (2012) for examples. In this paper, we develop, apply and investigate space-filling designs that take account of probabilistic prior information and, particularly, relationships between input variables. The eventual aim of the designs is to reduce the number of simulator evaluations required, compared to both (i) Monte Carlo sampling from the distributions for the, quantitative or qualitative, input variables; and (ii) standard space-filling designs. The designs are found using a new class of weighted space-filling criteria that are introduced in Section 2. Designs from competing criteria are evaluated and compared using assessment criteria introduced in Section 3 in terms of (a) sampling properties with respect to the prior information, and (b) space-filling properties. Illustrative examples are presented in Section 4 and the new methods are used to reduce the number of simulator evaluations needed for the dispersion problem in Section 5.

2. Weighted space-filling designs

Space-filling design methods are commonly used when the relationship between k inputs and the response is unknown but possibly highly complex and nonlinear. At any combination of input values, at most one observation is taken, and hence such designs are commonly used with deterministic simulators. For a recent review see Pronzato and Müller (2012). They have also found application in spatial statistics (for example, Royle, 2002).

Standard designs used in computer experiments include computer-generated minimax and maximin designs (Johnson et al., 1990), uniform designs (Fang, 1980) and Latin Hypercube designs (McKay et al., 1979). Here, we focus on extensions of coverage and spread designs (SAS Proc Optex, 1995), and in Section 4 we compare them to an appropriately generated Latin Hypercube design.

2.1. Coverage and spread designs

To apply space-filling designs to applications such as the dispersion simulator, we first need to define distance metrics across the design space. Let $x_1, \ldots, x_{k_1}$ denote k_1

quantitative variables and $x_{k_1+1}, \ldots, x_{k_1+k_2}$ denote k_2 unordered categorical variables, with variable x_j having m_j levels denoted by $\mathcal{M}_j = \{1, \ldots, m_j\}$ ($j = k_1 + 1, \ldots, k_1 + k_2$). Then, we can define the distance between two points $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X} = \mathcal{R} \times \prod_j \mathcal{M}_j$, where $\mathcal{R} \subset \mathbb{R}^{k_1}$, as

$$ d(\boldsymbol{x}, \boldsymbol{x}') = \sqrt{\sum_{i=1}^{k_1} (x_i - x_i')^2 + \beta \sum_{j=k_1+1}^{k_1+k_2} \mathbf{I}[x_j \neq x_{j)}']}, \tag{1} $$

where $\mathbf{I}[r \neq s]$ is the indicator function that takes the value 1 if $r \neq s$ and 0 otherwise. Equation (1) is a weighted sum (with respect to $\beta > 0$) of the L_2 distance for quantitative variables and the 0-1 distance for qualitative variables. Such distance measures were defined and applied to Gaussian process models by Qian et al. (2008).

Let $\zeta = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ be a n-point discrete design. Using (1), the following new weighted space-filling criteria can be defined:

Coverage criterion: a coverage-optimal design $\zeta^\star$ minimises

$$ \phi_u(\zeta) = \left\{ \int_{\mathcal{X}} \left[\min_{\boldsymbol{x} \in \zeta} w(\boldsymbol{x}') d(\boldsymbol{x}, \boldsymbol{x}') \right]^p \, \mathrm{d}\boldsymbol{x}' \right\}^{1/p}. \tag{2} $$

Spread criterion: a spread-optimal design $\zeta^\star$ minimises

$$ \phi_s(\zeta) = \left\{ \sum_{i=1}^{n} \left[\min_{\boldsymbol{x} \in \zeta \backslash \{\boldsymbol{x}_i\}} w(\boldsymbol{x}) w(\boldsymbol{x}_i) d(\boldsymbol{x}, \boldsymbol{x}_i) \right]^{-p} \right\}^{1/p}. \tag{3} $$

Here, $w(\boldsymbol{x}) \geq 0$ is a problem-specific multivariate weight function that encapsulates the dependencies between the input variables and which may or may not be a proper probability density function. In many applications, including the motivating dispersion setting, it can be thought of as the probability of obtaining a "useful" or representative response. We discuss the choice of $w(\cdot)$ further in Sections 4 and 5. The tuning parameter p allows a range of criteria with different properties to be defined; in this paper, we set $p = 1$.

For $w(\boldsymbol{x}) \equiv 1$ for all $\boldsymbol{x} \in \mathcal{X}$, these criteria are closely related to the minimax and maximin space-filling criteria defined by Johnson et al. (1990). A minimax design minimises through choice of design the *maximum*, over $\mathcal{X}$, of the distance $\min_{\boldsymbol{x} \in \zeta} d(\boldsymbol{x}, \boldsymbol{x}')$; a coverage design with $p = 1$ minimises the *average*, over $\mathcal{X}$, of this distance. By minimising the average rather than the maximum, we prevent worst-case distances unduly influencing the choice of design (see Atkinson and Woods, 2013 for a related discussion in the context of optimal model-based design). Clearly, the minimax and spread objective functions coincide in the limit as $p \to \infty$.

Similarly, a maximin design maximises $\min_{\boldsymbol{x} \in \zeta \backslash \{\boldsymbol{x}_i\}} d(\boldsymbol{x}, \boldsymbol{x}_i)$ through choice of design, whereas a spread design, with $p = 1$, minimises the *average* of the reciprocal of this minimum distance. In fact, it maximises the harmonic mean distance between design points.

Mathematically, for coverage designs, we want to attract the points in the design to relevant areas of the design region. Note that if $w(\boldsymbol{x}') = 0$, $\min_{\boldsymbol{x} \in d} w(\boldsymbol{x}') d(\boldsymbol{x}, \boldsymbol{x}') = 0$

for all choices of design ζ. Hence adding the point $\boldsymbol{x}'$ to will not enhance the coverage properties of the design. For spread designs, we want the points in the design to repel away from each other. Note that as $w(\boldsymbol{x}') \to 0$,

$$\left[\min_{\boldsymbol{x} \in d \setminus \{\boldsymbol{x}'\}} w(\boldsymbol{x}) w(\boldsymbol{x}') d(\boldsymbol{x}, \boldsymbol{x}') \right]^{-p} \to \infty$$

and hence point $\boldsymbol{x}'$ can never be included in the design.

In contrast to these tailored space-filling designs, random sampling from the underlying distribution on $\mathcal{X}$ risks unnecessary oversampling from regions of high density. In Section 5, we quantify the differences in simulation results obtained from the use of weighted space-filling designs and random sampling.

2.2. Implementation

Optimal and efficient designs under the coverage and spread criteria can be found using a variety of computer search algorithms. The designs in this paper were obtained using a modified-Fedorov (MF) point exchange algorithm (Cook and Nachtsheim, 1980; Royle, 2002). Regardless of the method employed, there are two key steps in finding a design

1. Evaluation of the objective function: to find spread designs, (3) can be evaluated directly. For coverage designs, (2) can be approximated using a quasi-random sequence $(\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_s) \in \mathcal{X}$ (see, for example, Lemieux, 2009, ch.5)

$$\phi_u(\zeta) \approx \sum_{j=1}^{s} \left\{ \left[\min_{\boldsymbol{x} \in \zeta} w(\boldsymbol{x}'_j) d(\boldsymbol{x}, \boldsymbol{x}'_j) \right]^p \right\}^{1/p} . \tag{4}$$

   For the illustrative examples in Section 4, we used $s = 900$ (Example 1, with two quantitative variables) and $s = 2700$ (Example 2, with two quantitative factors and one qualitative factor); for the dispersion application (Section 5), we used $s = 1000$. Alternatively, a quadrature scheme (see, for example, Evans and Swartz, 2000) could be employed.

2. Exploring the design space and improving the design: the MF algorithm proceeds by considering each point in the current design in turn and evaluating the objective function when this point is replaced by a point from a candidate list of possible design points. If the swap improves the objective function, then the candidate point immediately replaces the design point, and a swap with the next point in the candidate list is evaluated. Alternatively, swaps, or a continuous optimisation step, could be carried out on each coordinate of each design point (Meyer and Nachtsheim, 1995; Gotwalt et al., 2009).

   The MF algorithm is greedy, in that any swap that improves the current design is accepted. Stochastic algorithms, which attempt to avoid local optima of the objective function by randomly accepting changes to the design which reduce performance, could also be employed; examples include adaptive evolutionary algorithms (Wong and Dong, 2005). Implementation of these alternatives did not result in

an improvement to the generated designs, despite the more global nature of the optimization.

Using a candidate-list method, such as a row exchange algorithm, allows the pre-computation of the distances between all pairs of points. With other, more continuous, algorithms, efficient updating of distances is possible (as in the `Fields` package in `R`; Nychka, 2005).

We construct our candidate list using the same quasi-random sequence used tin approximation (4).

Clearly, the optimization methods discussed above are heuristic and have no guarantee of converging to a global optimum. For a greedy algorithm, such as the MF, it is therefore particularly important to perform multiple runs of the algorithm from different, random, starting locations. In this paper, it was judged sufficient, from the consistency of the generated designs, to perform 20 random starts.

3. Graphical assessment of designs

We graphically assess the properties of the design for both space-filling and sampling (with respect to the prior density) using the following assessment measures:

(i) Fraction of Design Space (FDS; Zahran et al., 2003) with respect to the distance (1). That is, for each point $\tilde{\boldsymbol{x}}$ in the design space, we calculate

$$\phi(\tilde{\boldsymbol{x}}|\zeta) = \min_{\boldsymbol{x}\in\zeta} d(\boldsymbol{x}, \tilde{\boldsymbol{x}}),$$

for $\tilde{\boldsymbol{x}} \in \mathcal{X}$ and plot the inverse of the empirical distribution function

$$\Phi_1(\nu|\zeta) = \frac{1}{\mathcal{D}} \int_{\mathcal{A}_1} \mathrm{d}\tilde{\boldsymbol{x}}, \tag{5}$$

where $\mathcal{A}_1 = \{\tilde{\boldsymbol{x}} \in \mathcal{X} | \phi(\tilde{\boldsymbol{x}}|\zeta) \leq \nu\}$, $\mathcal{D} = \int_{\mathcal{X}} \mathrm{d}\boldsymbol{x}$ and $0 \leq \Phi_1(\nu|\zeta) \leq 1$ for all $\nu \geq 0$. Intuitively, $\Phi_1(\nu|\zeta)$ has a straightforward interpretation as the proportion of the design space $\mathcal{X}$ that is within ν of design ζ; that is, the proportion of the design space for which $\phi(\tilde{\boldsymbol{x}}|\zeta) \leq \nu$ for a point $\tilde{\boldsymbol{x}}$.

Clearly, this assessment measure is "larger-the-better", and design ζ_1 dominates design ζ_2 if and only if $\Phi_1(\nu|\zeta_1) \geq \Phi_1(\nu|\zeta_2)$ for all ν, with $\Phi_1(\nu|\zeta_1) > \Phi_1(\nu|\zeta_2)$ for at least one value of ν.

We approximate the integral in (5) across $\mathcal{A}_1$ for any given ν by a summation across a quasi-random sample $\tilde{x}_1, \ldots, \tilde{x}_r$ from $\mathcal{X}$

$$\phi_1(\nu|\zeta) \approx \frac{1}{r} \sum_{j=1}^{r} \mathbf{I}[\phi(\tilde{\boldsymbol{x}}_j|\zeta) \leq \nu],$$

where $\mathbf{I}$ is once again an indicator function. Clearly, too small a quasi-random sample may lead to inaccurate design assessment and, most importantly, incorrect ranking of designs. We used $r = 1000$ and found the assessment and ranking of designs to be consistent across different samples.

(ii) *Fraction of Design Points (FPS)* with respect to the sampling density $p(\boldsymbol{x})$. That is, for each point $\boldsymbol{x}$ in the design, we calculate $p(\boldsymbol{x})$ and then plot the inverse of the empirical distribution function

$$\Phi_2(\rho|\zeta) = \frac{1}{n} \int_{\mathcal{A}_2} d\tilde{\boldsymbol{x}}, \tag{6}$$

where $\mathcal{A}_2 = \{\tilde{\boldsymbol{x}} \in \zeta | p(\tilde{\boldsymbol{x}}) \leq \rho\}$ and $0 \leq \Phi_2(\rho|\zeta) \leq 1$ for all $\rho \geq 0$. For this measure, for a given ρ, smaller values of $\Phi_2(\rho|\zeta)$ are intuitively desirable, as then a higher proportion of design points have higher values of $p(\boldsymbol{x})$.

Clearly, this assessment measure is "smaller-the-better", and design ζ_1 dominates design ζ_2 if and only if $\Phi_2(\rho|\zeta_1) \leq \Phi_2(\rho|\zeta_2)$ for all ρ, with $\Phi_2(\rho|\zeta_1) < \Phi_2(\rho|\zeta_2)$ for at least one value of ρ.

For designs with a discrete and finite number of points, as in this paper, (6) can be calculated as

$$\Phi_2(\rho|\zeta) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{I}[p(\boldsymbol{x}_i) \leq \rho].$$

These two assessment measures can be interrogated graphically to compare and select designs, as demonstrated in the next two sections.

4. Illustrative examples

In this section we consider two relatively simple examples which illustrate different properties of the criteria from Section 2. We start by defining two different weight functions.

4.1. Weight functions
Assuming a prior distribution, $p(\boldsymbol{x})$, on $\mathcal{X}$ can be elicited from subject experts. We consider the following weight functions:

$$w(\boldsymbol{x}) = p(\boldsymbol{x}), \quad w(\boldsymbol{x}) \geq 0, \tag{7}$$

and

$$w(\boldsymbol{x}) = (1 - \alpha p(\boldsymbol{x}))^{-\gamma}, \quad w(\boldsymbol{x}) \geq 1, \tag{8}$$

with tuning parameters $\alpha < 1/\max p(\boldsymbol{x})$ and $\gamma \geq 0$. The inverse of weight function (8) was used by Joseph et al. (2011) in the sequential design of physical experiments in nanoscience, and α and γ allow the weight function to be adapted to the aim of the experiment by putting greater emphasis on sampling or space-filling. For example, when $\alpha = 0$ or $\gamma = 0$, the weight function is constant and space-filling coverage or spread designs result.
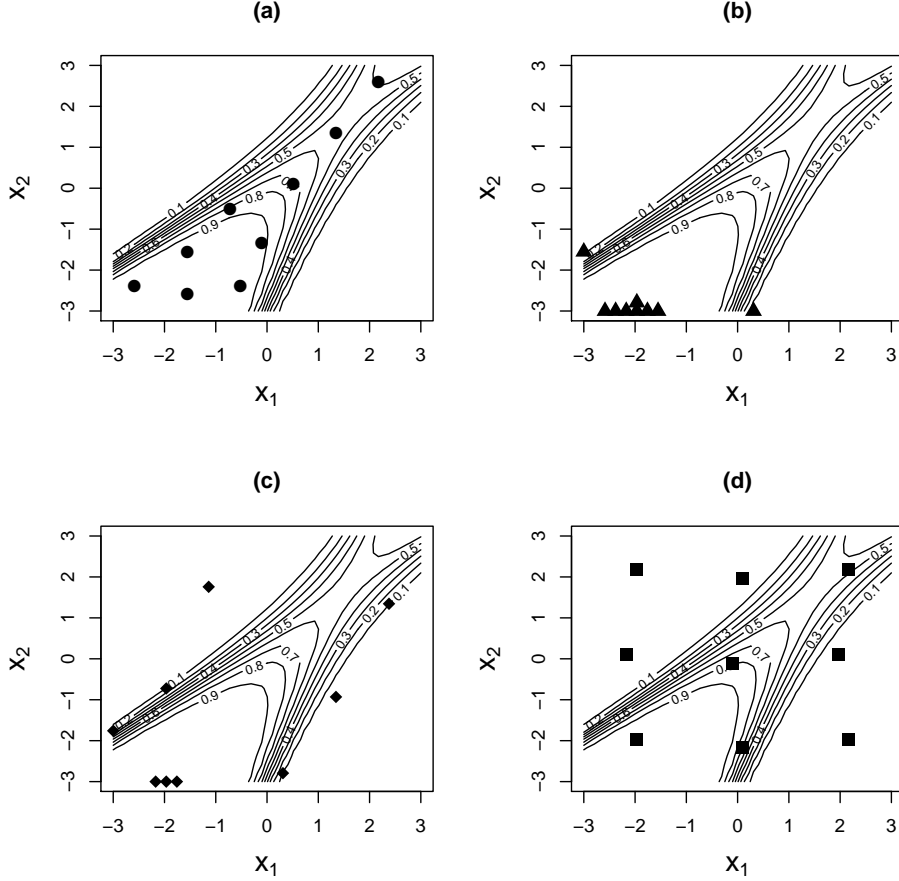
Figure 3: Example 1: Coverage designs. (a) ζ_{c1}, weight function (7); (b) $\zeta_{c2}(1)$, weight function (8) with $\alpha = \gamma = 1$; (c) $\zeta_{c2}(0.75)$, weight function (8) with $\alpha = 1$, $\gamma = 0.75$; (d) ζ_c, unweighted design.

4.2. Example 1

The first illustrative example has two quantitative variables, $\boldsymbol{x} = (x_1, x_2)$, and $p(\boldsymbol{x})$ defined via the logistic function

$$\ln \frac{p(\boldsymbol{x})}{1 - p(\boldsymbol{x})} = 1.2 + 0.7x_1 - 1.8x_2 - 1.9x_1x_2 - 0.8x_1^2 + 3.0x_2^2, \tag{9}$$

for $\boldsymbol{x} \in \mathcal{X} = [-3, 3]^2$. Hence, the prior density $p(\boldsymbol{x})$ is a ridge function with substantial regions having $p(\boldsymbol{x}) \approx 0$, see Figure 3. We find space-filling designs with nine points.

Standard coverage (ζ_c; Figure 3d) and spread designs (ζ_s; Figure 5d) in two dimensions assume $w(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \mathcal{X}$ and hence attempt to provide uniform space-filling across $\mathcal{X}$. Note that the spread design has points closer to the boundary of $\mathcal{X}$. As they take no account of the weight function, both designs have many points in areas of low weight.

Weighted coverage and spread designs were found using an exchange algorithm to minimise objective functions (2) and (3) respectively using weight functions (7) and (8). The designs are shown in Figures 3 (coverage) and 5 (spread).
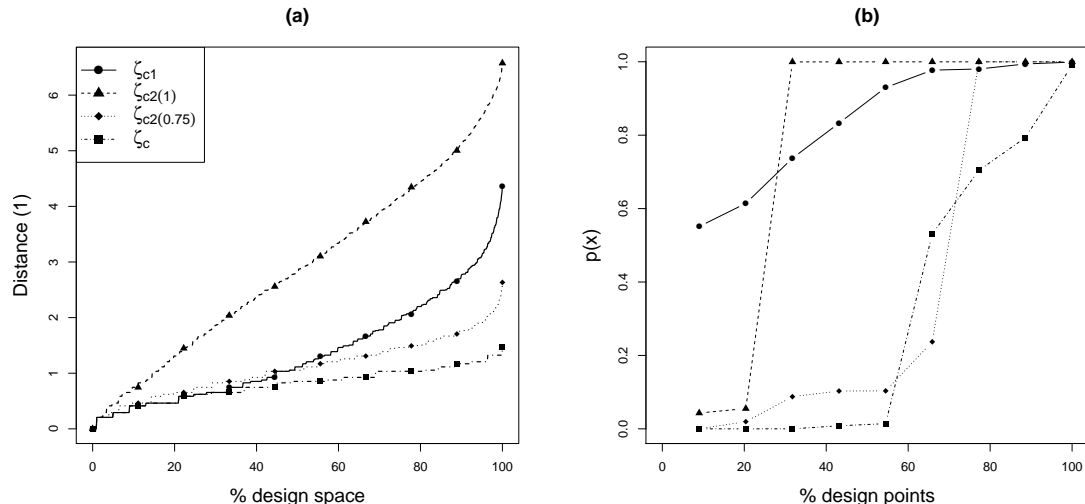
9

Figure 4: Example 1: (a) Fraction of Design Space and (b) Fraction of Design Point plots for the coverage designs.

The design found using (7), ζ_{c1}, places all nine design points on the diagonal ridge of high probability; the minimum euclidean distance between the points is 1.03. Two designs, $\zeta_{c2}(1)$ and $\zeta_{c2}(0.75)$, were found using (8), $\alpha = 1$, and $\gamma = 1$ and $\gamma = 0.75$. These designs include points which are clustered in the area of highest prior probability. Both designs include points between which there is a Euclidean distance of 0.21, the minimum distance within the limitations of the candidate list. In addition, both designs have points in areas of lower prior density, with more points in areas of even lower density for $\zeta_{c2}(0.75)$ where greater emphasis is placed on space-filling.

For Example 1, graphical assessment via FDS and FDP is possible via Figure 4. Note that we plot $\phi_1^{-1}(\nu|\zeta)$ and $\phi_2^{-1}(\rho|\zeta)$, and hence in the FDS and FDP plots, we prefer lower and higher curves respectively. The designs ζ_{c1} and $\zeta_{c2}(1)$ dominate both the weighted $\zeta_{c2}(0.75)$ and unweighted ζ_c designs in the FDP plot; $\zeta_{c2}(1)$, found using weight function (8) has seven points obtaining the maximum $p(\boldsymbol{x})$. However, in the FDS plot, $\zeta_{c2}(1)$ is dominated by all the three other designs. In contrast, ζ_{c1}, found using weight function (7), is dominated only by the unweighted coverage design. Hence, ζ_{c1} provides a good compromise between space-filling and sampling.

The spread designs display similar patterns to the coverage designs, see Figure 5. However the distribution of points tends to be more extreme under weight function (8), with design $\zeta_{s2}(1)$ having all nine points closely clustered whereas design $\zeta_{s2}(0.5)$ has design points at the boundary of the design region. When assessed via FDP and FDS measures (plots not shown for these designs), design $\zeta_{s2(1)}$ dominates all other designs on FDP but is itself dominated by the other three designs on FDS. Interestingly, the unweighted spread design ζ_s does not dominate ζ_{s1} and $\zeta_{s2}(0.5)$ under FDS. This is because FDS assesses the coverage properties of a design, and d_s is found under the spread criterion.

To compare the coverage and spread designs, Figure 6 repeats the FDP and FDS plots for designs ζ_{c1}, ζ_{s1}, ζ_s and ζ_c. The weighted coverage and spread designs ζ_{c1} and ζ_{s1}
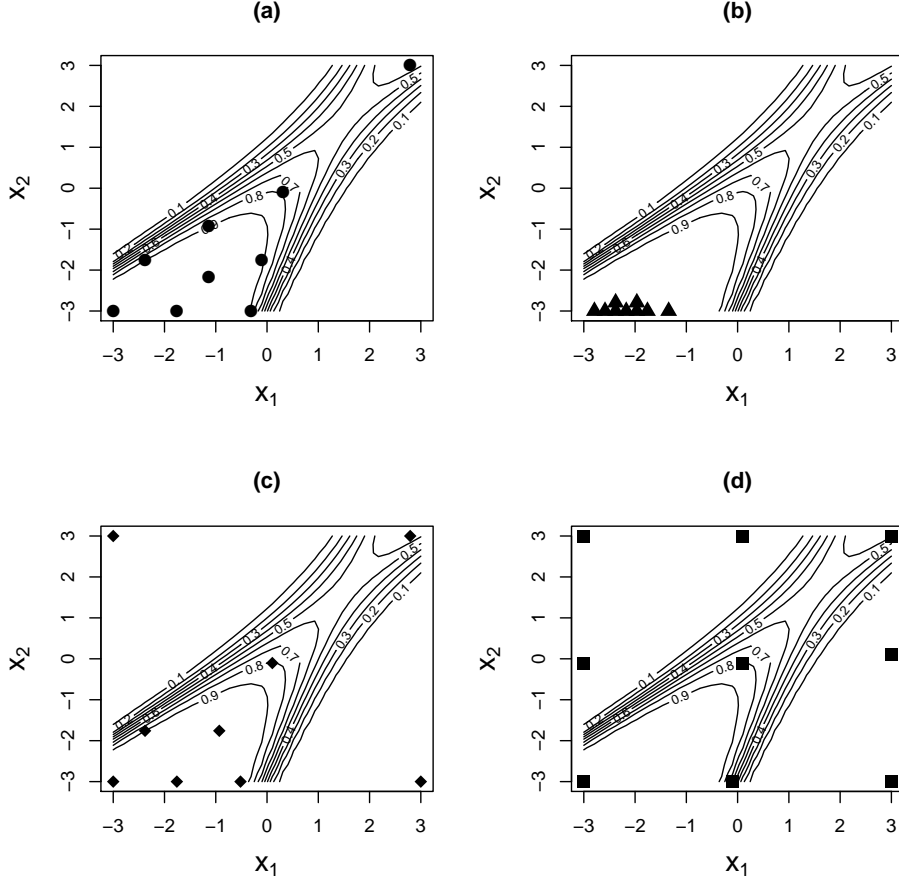
10

Figure 5: Example 1: Spread designs. (a) ζ_{s1}, weight function (7); (b) $\zeta_{s2}(1)$, weight function (8) with $\alpha = \gamma = 1$; (c) $\zeta_{s2}(0.5)$, weight function (8) with $\alpha = 1$, $\gamma = 0.5$; (d) ζ_s unweighted design.

have similar FDP plots but ζ_{c1} dominates ζ_{s1} in the FDS plot. Similarly, it is difficult to separate ζ_c and ζ_s in the FDP plot but ζ_c dominates all other designs in the FDS plot. Of course, we do not expect the spread designs to outperform the coverage designs under the FDS criterion.

4.3. Example 2

The second illustrative example has three variables $\boldsymbol{x} = (x_1, x_2, x_3)$; two quantitative variables ($x_1, x_2 \in [-3, 3]^2$) and one qualitative variable ($x_3 \in \{0, 1, 2\}$). Conditional on the value of x_3, $p(\boldsymbol{x}|\Sigma)$ is a bivariate normal probability density function

$$p(\boldsymbol{x}|\Sigma) = \frac{1}{2\pi \mid \Sigma \mid^{1/2}} \exp\left\{ -\frac{1}{2}[\boldsymbol{x} - \boldsymbol{\mu}(x_3)]'\Sigma^{-1}[\boldsymbol{x} - \boldsymbol{\mu}(x_3)] \right\} ,$$
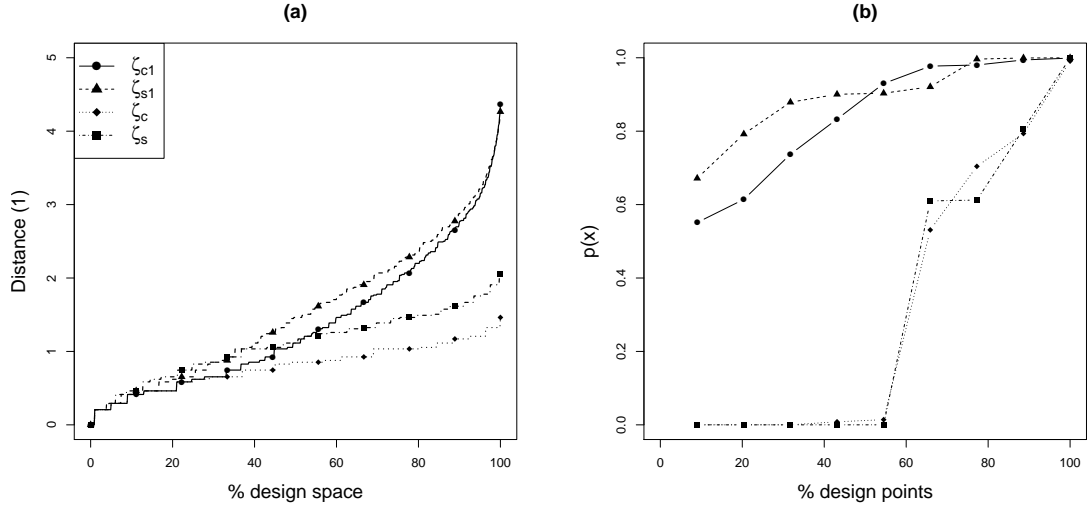
with

11

Figure 6: Example 1: (a) Fraction of Design Space and (b) Fraction of Design Point plots for the coverage and spread designs.

$$
\mu(x_3) = \left\{
\begin{array}{ll}
(0, 0)' & \text{if } x_3 = 0 \\
(1, 1)' & \text{if } x_3 = 1 \\
(-1, -1)' & \text{if } x_3 = 2,
\end{array}
\right.
$$

and $\Sigma = I_2 + J_2$. Here I_2 is the 2×2 identity matrix and J_2 is the 2×2 matrix with all entries 1. This density function has ellipsoidal contours in (x_1, x_2), with a different centre for each level of x_3; see Figure 7, which also displays weighted coverage and spread designs found using (7).

Coverage and spread designs were found for both weight function (7) (ζ_{c1} and ζ_{s1}; Figure 7) and (8) (ζ_{c2} and ζ_{s2}; not shown). For weight function (8), a variety of values of α and γ were empirically investigated, with the coverage design with $\alpha = 1/\max\{p(\boldsymbol{x}|\Sigma)\} = 0.37$, $\gamma = 0.75$ and the spread design with $\alpha = 0.37$ and $\gamma = 0.3$ giving good compromises between sampling and space-filling.

As $p(x_3) = 1/3$ for $x_3 = 0, 1, 2$, all designs have three of their nine points at each of the levels of x_3. The coverage designs have points clustered around the centre of each projection of $p(\boldsymbol{x}|\Sigma)$ in such a way that the projection of all nine points into the $x_1 - x_2$ plain maintains good coverage (Figure 7 shows the projection for $x_3 = 0$). Design ζ_{c1} is more concentrated than ζ_{c2} around regions of high probability density, with ζ_{c2} containing a few points in areas of very low density.

The spread design, ζ_{s1}, from weight function (7), has a similar distribution of points to the coverage designs, with clustering around the centres of each projected density. As in Example 1, both the coverage and spread designs from (8) provide a more extreme compromise between sampling and space-filling. For the spread design ζ_{s2} under weight function (8), the majority of the design points are closer to the three unique modes for $x_3 = 0, 1, 2$. However, two points are positioned at the extremes of the design region, in areas where $p(\boldsymbol{x}|\Sigma)$ is close to 0. Although these points greatly increase the performance of the design under a spread criterion, they provide little in terms of sampling efficiency.
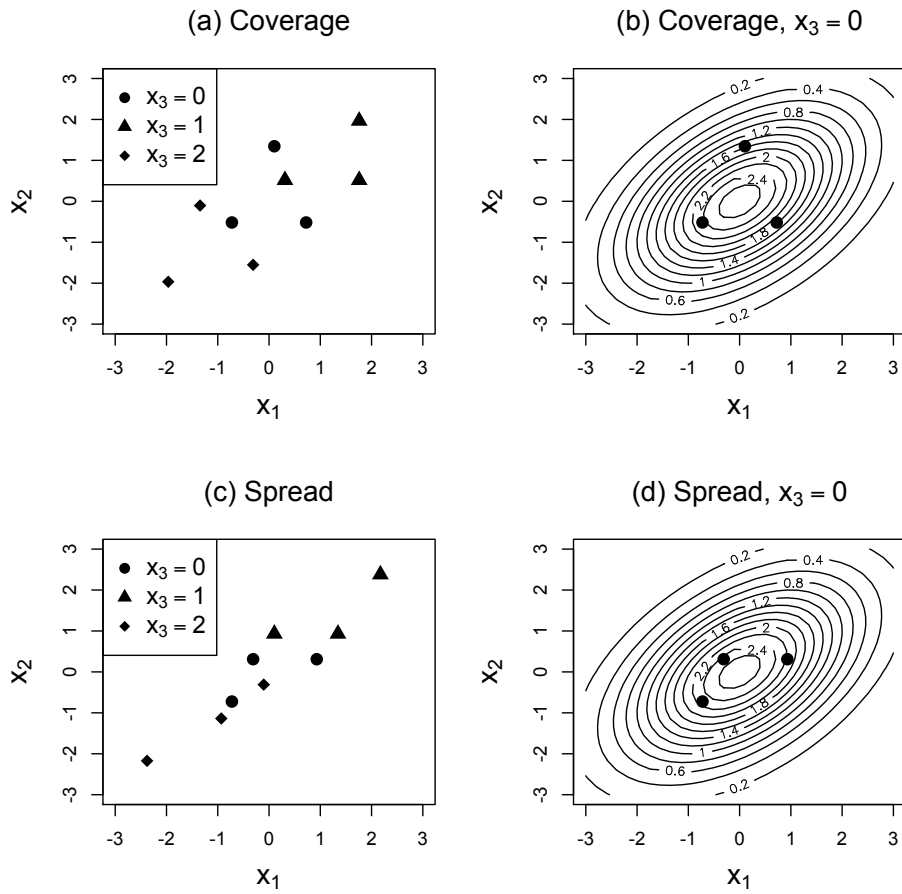
12

Figure 7: Example 2: Coverage, ζ_{c1}, and spread, ζ_{s1}, designs using weight function (7). (a) design points for ζ_{c1}; (b) design points and weight function (7) for ζ_{c1} when $x_3 = 0$; (c) design points for ζ_{s1}; (d) design points and weight function (7) for ζ_{s1} when $x_3 = 0$.
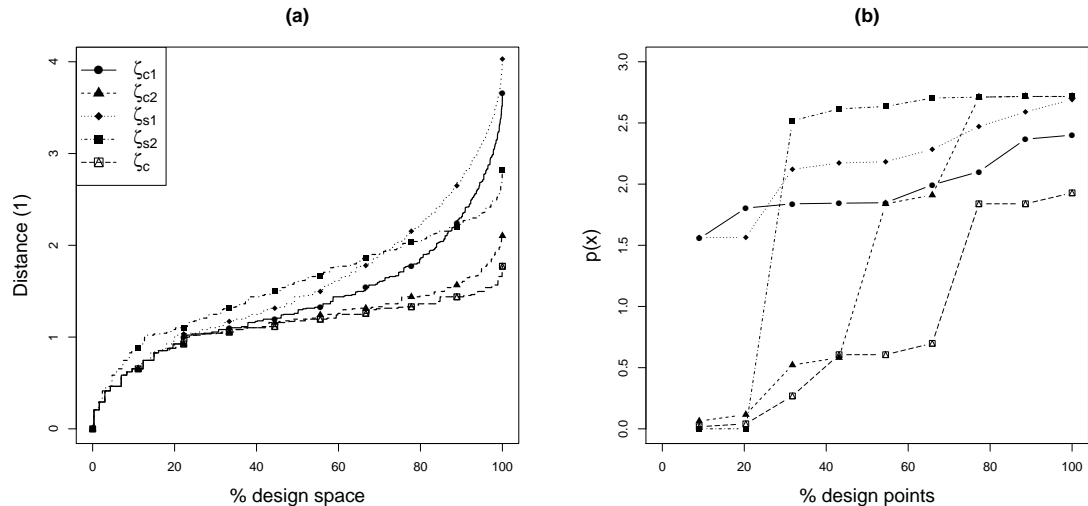
Figure 8: Example 2: (a) Fraction of Design Space and (b) Fraction of Design Point plots for the coverage and spread designs.

We have confirmed through the construction of alternative designs that the level of x_3 for these extreme points is unimportant, providing there are three points at each level of the variable.

We can again assess and compare the properties of these different designs using FDP and FDS plots (Figure 8). No one design dominates in the FDP plot, although the coverage (ζ_{c1}) and spread (ζ_{s1}) designs found using weight function (7) are the only designs that avoid placing any points in areas of very low probability. Design ζ_{s1} is generally preferred over ζ_{c1} from the FDP plot; however, ζ_{c1} dominates ζ_{s1} in the FDS plot. However, both these designs are dominated by ζ_{c2} under FDS. In fact, design ζ_{c2} performs similarly under the FDS criterion to the unweighted coverage design, ζ_c, whilst substantially outperforming ζ_c under FDP. As in Example 1, these measures and graphical displays can be used to choose between these, and other, designs depending on the aim of the experiment and the confidence in the prior probability density.

5. Application to dispersion

We now return to the motivating dispersion example to find and assess appropriate weighted coverage and spread designs for dispersion across a flat, rural (non-urban) inland terrain which is subject to particularly skewed meteorological conditions due to the terrain surrounding the area of interest. The dispersion model is a massively multivariate simulator, with each simulator run producing 16384 responses, one response for each point on the 128×128 grid.

To apply a weighted space-filling design to a practical example, three basic steps are required: (i) the simulator inputs to be varied in the experiment must be chosen; (ii) prior information about these variables and their dependencies must be obtained, for example, from historical data or elicited from subject experts; and (iii) a size of design must be chosen.

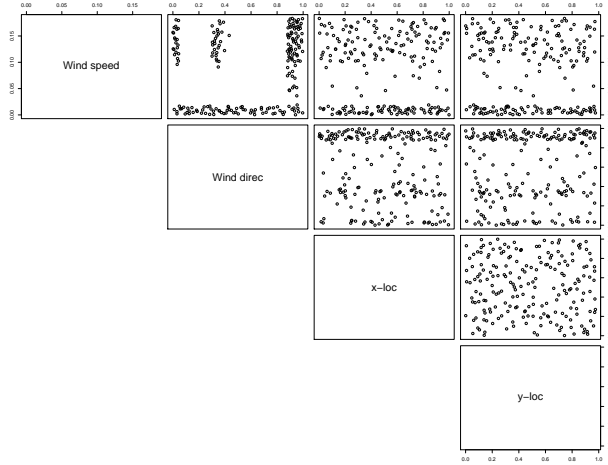In order to explore the methodology we considered two examples using different sub-

14

Figure 9: Example 3: selected two-dimensional projections of the weighted coverage design ζ_{c1}^7.

sets of input variables. The first set of input variables $\mathbf{x}^7$ are a set of the seven most significant quantitative variables: wind speed, wind direction, cloud cover, temperature, two Cartesian coordinates giving the location of the release, and the mass of the release. The second subset $\mathbf{x}^{10}$ contain the same set of quantitative variables but with the three most significant categorical variables appended: dispersion mechanism (7 levels), release-type (3 levels) and type of agent (5 levels). When finding space-filling designs, each variable was scaled from its original range to $(0, 1)$.

For both examples, the prior distributions for wind speed and wind direction were non-uniform, with the dependency between these variables estimated from historical data. This relationship was encapsulated in a two-way table for six values of speed and 13 for direction; the probability density for arbitrary points in the design space calculated via interpolation between the entries in this table. Uniform distributions were assumed for cloud cover and release location, and a normal distribution was assumed for temperature. The distribution for mass varied between the two examples, with the second example also requiring specification of prior distributions for the three categorical variables. The joint density for both the seven and ten variable examples reflected realistic dispersion scenarios in terms of historical data and subject-expert opinion; see Appendix 1. For both examples, weight function (7) was used.

5.1. Example 3: quantitative variables only

For this example, a four point discrete probability distribution for mass was elicited from subject experts. Space-filling designs, ζ_{c1}^7 and ζ_{s1}^7, using coverage (2) and spread (3) respectively were obtained with 200 points using the prior density derived above; un-weighted coverage (ζ_c^7) and spread (ζ_s^7) designs were also found. Selected two-dimensional projections for ζ_{c1}^7 are given in Figure 9. The strong relationship between wind speed and direction is reflected in the selected design points. Notice also the independence between the two Cartesian location coordinates.

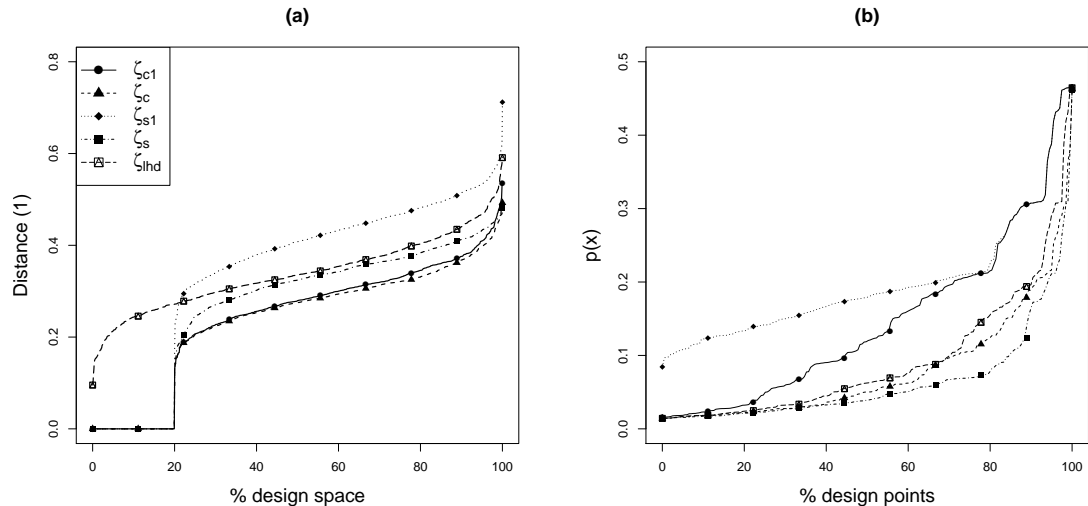For comparison, a 200 point Latin Hypercube Design (LHD; ζ_{lhd}) was also generated.

Figure 10: Example 3: (a) Fraction of Design Space and (b) Fraction of Design Points plots for weighted coverage (ζ_{c1}^7) and spread (ζ_{s1}^7) designs, unweighted coverage (ζ_c^7) and spread (ζ_s^7), and a Latin Hypercube Design (ζ_{lhd}).

An LHD with n points is a form of stratified sampling where the range of each input variable is delineated into n bins of equal probability with respect to the marginal distribution of the variable. Typical choices of marginal distribution include the Uniform and Normal distributions. Often, the n values of each variable are permuted, or the design otherwise constructed, to obtain desirable two-dimensional properties, such as orthogonality. We form the design by permuting the n variable values to best match the correlation structure of our target distribution using the method of Iman and Conover (1982).

The coverage, spread and Latin Hypercube designs were assessed using the FDP and FDS criteria (Figure 10). For FDP, the weighted spread design ζ_{s1}^7 dominates all the other designs, with the weighted coverage design ζ_{c1}^7 dominating the LHD and the two unweighted space-filling designs. Design ζ_{c1}^7 has 40% of its points having probability density less than 0.1; for ζ_{s1}^7, only 5% of points have density less than 0.1. For FDS, the unweighted coverage design ζ_c^7 dominates, although the difference between this design and ζ_{c1}^7 is only small. The weighted spread design is considerably worse, having 50% of the design space having a minimum distance of more than 0.4 units from a design point; for design ζ_{c1}^7, only about 10% of the design space has a minimum distance more than 0.4 units from a design point.

From Figure 10, it is clear that while ζ_{c1}^7 provides more points in areas of low probability density than ζ_{s1}^7, it has considerable more desirable space-filling properties. Hence, we chose ζ_{c1}^7 for a computer experiment on the dispersion model. The LHD is mediocre for both FDP and FDS but was also run as a comparator. Finally, a 200 Monte Carlo sample from the prior probability distribution was also obtained and run through the dispersion model.

For a quantitative comparison of these three designs, ζ_{c1}^7, ζ_{lhd}^7 and the 200 run Monte Carlo sample, we calculated, at each point in the 128×128 grid, the squared error for

16

Table 1: Example 3: Squared error summaries from a comparison of the dispersion surface from an 80,000 run Monte Carlo simulation to that from three competing designs: weighted coverage (ζ_{c1}^{7}), Latin Hypercube (ζ_{lcd}^{7}) and a Monte Carlo sample of size 200.

Squared error	Mean	St. Dev.	Max.
ζ_{c1}^{7}	4.0×10^{-4}	7.0×10^{-4}	5.4×10^{-3}
ζ_{lhd}^{7}	6.0×10^{-4}	7.0×10^{-4}	5.0×10^{-3}
Monte Carlo ($n = 200$)	7.0×10^{-3}	8.5×10^{-3}	4.1×10^{-2}

the proportion of plumes observed obtained from the simulator runs from each design, relative to the proportion of observed plumes from a computationally expensive Monte Carlo sample of size 80,000. These results are summarised in Table 1. Both ζ_{c1}^{7} and ζ_{lhd}^{7} have mean and maximum squared error an order of magnitude smaller than the Monte Carlo sample. Design ζ_{c1}^{7} has mean squared error only two-thirds that of ζ_{lhd}^{7}, with maximum squared error less than 10% greater. Both designs have the same variation in the squared error. A further analysis showed that a mean Monte Carlo sample of at least double the size of ζ_{c1}^{7} is required to achieve a similar mean squared error.

5.2. *Example 4: quantitative and categorical variables*

For this example, bivariate distributions were defined for dispersion mechanism and mass, dispersion mechanism and agent, and dispersion mechanism and release, see Appendix 1. Weighted space-filling designs, ζ_{c1}^{10} and ζ_{s1}^{10}, using coverage (2) and spread (3) respectively, were obtained with 200 points; unweighted coverage (ζ_{c}^{10}) and spread (ζ_{s}^{10}) designs were also found. To reduce the computational burden of the design search, the three categorical factors were treated as quantitative, and the selected values for each design then binned to into categories. However, when evaluating the designs, distance (1) was used with $\beta > 0$.

Figure 11 gives the FDS and FDP plots for these weighted and unweighted space-filling designs, with FDS evaluated using $\beta = 1, 10, 100$. For FDP, the weighted coverage design ζ_{c1}^{10} dominates the three other designs, which perform similarly. The skewed nature of the prior distribution on the larger design space for this example produce FDP plots that are quite flat for a majority of the design points ($> 70\%$). For FDS, the unweighted coverage design ζ_{c}^{10} dominates, although the differences between designs tend to be small. Note that just less than 20% of the design space is very close to a design point. Changing the value of β for design evaluation does not change the relative performance of the designs but clearly does increase the maximum distance, as the influence of combinations of the categorical variables not included in the design becomes greater.

6. Discussion

When there are probabilistic relationships and dependencies between input variables to a computer model or simulation that define regions of varying interest to the experimenters, standard space-filling designs may be inefficient and result in wasted resource. The methods in this paper allow the flexible construction of space-filling designs that incorporate such dependencies and prior information through a weight function, and hence
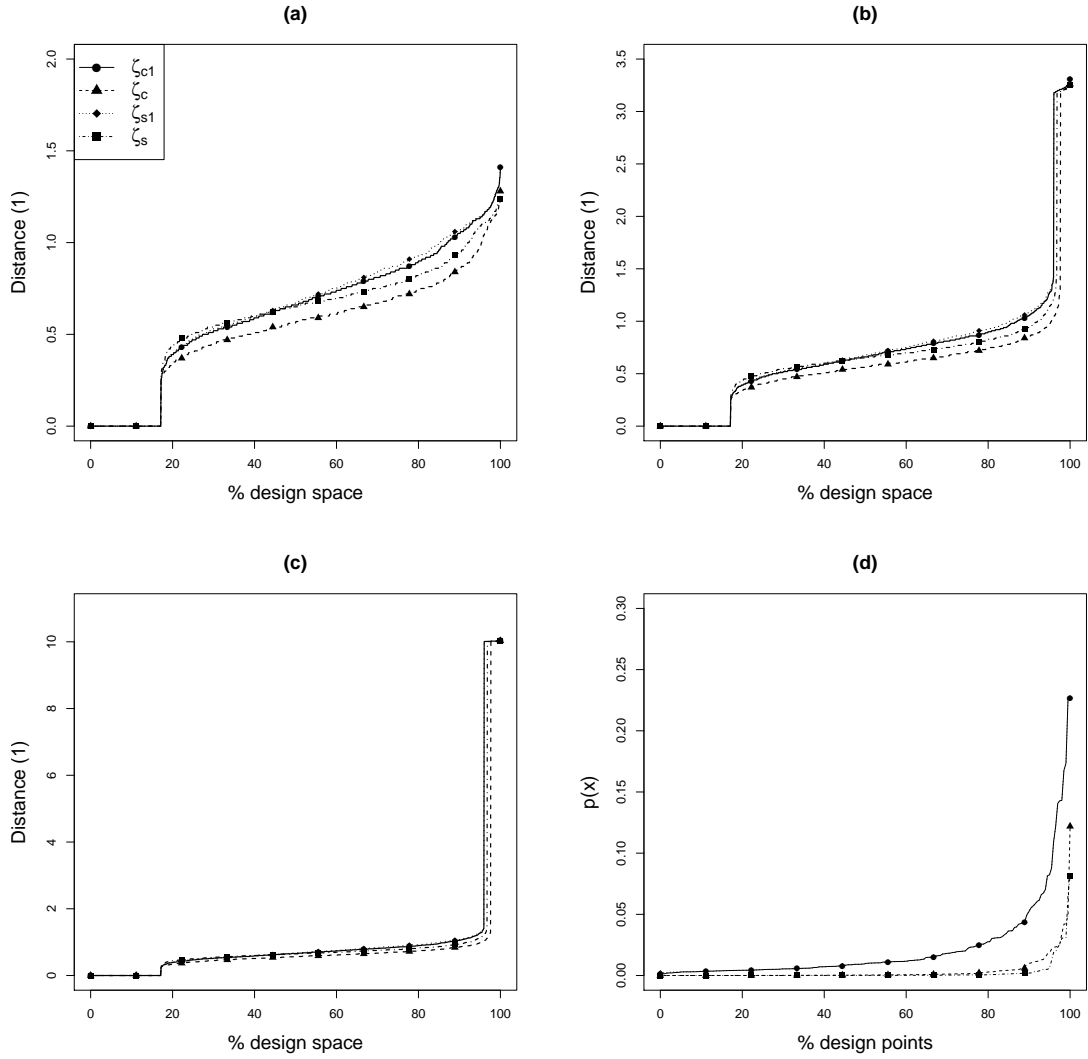
Figure 11: Example 4: Fraction of Design Space plots with (a) $\beta = 1$, (b) $\beta = 10$ and (c) $\beta = 100$, and (d) Fraction of Design Points plot for weighted coverage (ζ_{c1}^{10}) and spread (ζ_{s1}^{10}) designs, and unweighted coverage (ζ_{c}^{10}) and spread (ζ_{s}^{10}).

18

target the design towards relevant regions of the design space. The weight function employed can reflect the aims of the experiment, the nature of the prior information and the experimenter's confidence. Potentially, the weight function might reflect the utility of the design for some particular purpose, such as model-fitting. This is an interesting avenue for future research.

Two design selection criteria were investigated, coverage and spread. In general, based on evidence from the examples in this paper and other similar studies, we prefer the coverage criterion as it tends to provide more predictable and intuitive designs. When $w(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \mathcal{X}$, it also provides designs which seem better matched to the goals of space-filling. However, coverage designs are substantially more computationally expensive to find than spread designs, as their computation requires an approximation to an integral across the design space, see (2). Hence, for larger examples with more input variables, spread designs may be a more feasible option and, through choice of weight function, may still be tailored to the goals of the experiment.

For some applications, a simple inverse transformation of a standard space-filling design with respect to the weight function may be an alternative to the methodology presented here. However, in Section 5 we demonstrate that our weighted space-filling method still has benefits. In addition, for many practical problems including our motivating example, the weight function or prior distribution is only implicitly defined, making an inverse transformation difficult to achieve.

The methods also have potential application in spatial experiments, where a weight function may be constructed from prior knowledge or historical data on covariate values. For example, it may be of more interest to take observations in areas of high population density or low annual rainfall. The methods can also be directly applied to the selection of subsets of meteorological ensembles (Wilks, 2006, ch.6), where each ensemble member is weighted by the underlying meteorological distribution or where an empirical weight function is generated from the ensemble for the selection of further "pseudo-ensemble" members.

We have considered deterministic simulators. Designs for stochastic simulators, see Kleijnen (2008a,b), usually incorporate some repetition of combinations of simulator input variables, and so without modification, it is unlikely the coverage and spread criteria could be usefully applied. The utility of extensions to the methods developed in this paper for stochastic simulators could be investigated in the future.

Other areas of further work include the investigation of alternative distance metrics, such as the Mahalonobis distance for both quantitative and qualitative variables (Bedrick et al., 2000). Comparisons to samples from quasi-random numbers and low discrepancy sequences (see, for example, Lemieux, 2009) would also be insightful.

Space-filling in many dimensions with a small number of design points is clearly a difficult task. By weighting the design region according to experimenters' interest, we can reduce the effective size of the design space, making space-filling more feasible. In this sense, we suggest that weighting the design region has similar benefits to focussing on the space-filling of projections (see Lam et al., 2002).

Acknowledgements

A. Appendix 1

The following tables show the pairwise dependencies between the input variables for the two examples in Section 5.

Table 2: Two-way table of relative frequencies for wind speed and wind direction. Each cell denotes the percentage of time spent in the given ranges

		Wind Speed (m.s^{-1})					
		0.515	3.09	5.665	8.24	10.815	13.39
	16	1.78	0.6	0.1	0.1	0	0
	46	1.785	1.6	0.1	0.1	0	0
	76	1.785	3.9	0.3	0.1	0	0
	106	1.785	5.7	0.6	0.2	0	0.1
Wind Direction (°)	136	1.785	1.2	0.1	0	0	0.1
	166	1.825	1.8	0.2	0	0.1	0
	196	2.175	6.6	0.7	0.1	0	0
	226	2.175	2.7	0.3	0.1	0.1	0.1
	256	2.175	8.6	0.8	0.1	0.1	0
	286	2.175	30	4.9	0.3	0.1	0.1
	316	2.185	0.5	0.1	0.1	0.1	0
	346	2.185	0.1	0	0	0	0
	360	1.785	0.6	0.1	0.1	0	0

Table 3: Two-way table of relative frequencies for dispersion mechanism and mass. The dispersion mechanisms have been anonymised

		A	B	C	D	E	F	G
				Dispersion Mechanism				
Mass (kg)	10	100						40
	25						4	40
	50		75				4	20
	100		25		20		5	
	200			10	60		72	
	300			10	20	20	8	
	500			75		60	7	
	750			5		20		

Table 4: Two-way table of relative frequencies for dispersion mechanism and release type. Variables labels have been anonymised

		Release Type		
		Alpha	Beta	Gamma
Dispersion Mechanism	A	100	0	0
	B	100	0	0
	C	100	0	0
	D	100	0	0
	E	100	0	0
	F	0	80	20
	G	0	30	70

Table 5: Two way table of relative frequencies for dispersion mechanism and agent type. Variable labels have been anonymised

| | | Dispersion Mechanism | | | | | | |
		A	B	C	D	E	F	G
Agent Type	Mu	28	30	24.75	16.33	32	8.8	3
	Xi	28	30	24.75	16.33	32	8.8	3
	Omikron	28	30	24.75	16.33	32	8.8	2
	Pi	5.5	5	24.75	50	2	56.8	61
	Rho	10.5	5	1	1	2	16.8	31

References

Allen, T. T. (2011) *Introduction to Discrete Event Simulation and Agent-based Modeling.* New York: Springer.

Atkinson, A. C. and Woods, D. C. (2013) Designs for generalized linear models. In *Handbook of the Design and Analysis of Experiments* (eds. A. M. Dean, D. R. Bingham, M. D. Morris and J. Stufken), in press. Boca Raton FL: Chapman and Hall.

Bedrick, E. J., Lapidus, J. and Powell, J. F. (2000) Estimating the Mahalanobis distance from mixed continuous and discrete data. *Biometrics*, **56**, 394–401.

Clarke, R. H. (1979) The first report of a working group on atmospheric dispersion: A model for short and medium range dispersion of radionuclides released to the atmosphere. *Tech. Rep. NRPB-R91*, National Radiological Protection Board, Harwell.

Cook, R. D. and Nachtsheim, C. J. (1980) A comparison of algorithms for constructing exact *D*-optimal designs. *Technometrics*, **22**, 315–324.

De Rocquigny, E., Devictor, N. and Tarantola, S. (eds.) (2008) *Uncertainty in Industrial Practice: A Guide to Quantitative Uncertainty Management.* Hoboken, NJ: Wiley.

Draguljić, D., Santner, T. J. and Dean, A. M. (2012) Non-collapsing space-filling designs for bounded non-rectangular regions. *Technometrics*, **54**, 169–178.

Evans, M. and Swartz, T. (2000) *Approximating Integrals via Monte Carlo and Deterministic Methods.* Oxford: Oxford University Press.

Fang, K., Li, R. and Sudjianto, A. (2006) *Design and Modelling for Computer Experiments.* Boca Raton: Chapman and Hall.

Fang, K. T. (1980) The uniform design: application of number-theoretic methods in experimental design. *Acta Mathematicae Applicatae Sinica*, **3**, 363–372.

Gotwalt, C. M., Jones, B. A. and Steinberg, D. M. (2009) Fast computation of designs robust to parameter uncertainty for nonlinear settings. *Technometrics*, **51**, 88–95.

Iman, R. L. and Conover, W. J. (1982) A distribution-free approach to inducing rank correlation among input variables. *Communications in Statistics - Simulation and Computation*, **11**, 311–334.

Johnson, M. E., Moore, L. M. and Ylvisaker, D. (1990) Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, **26**, 131–148.

Jones, A. R., Thomson, D. J., Hort, M. and Devenish, B. (2007) The UK Met Office's next-generation atmospheric dispersion model, NAME III. In *Air Pollution Modeling and its Application XVII* (eds. C. Borrego and A.-L. Norman), 508–589. New York: Springer.

Joseph, V. R., Dasgupta, T. and Wu, C. F. J. (2011) Minimum energy designs: from nanostructure synthesis to sequential opimization. *Submitted for publication.*

Kennedy, M. C., Anderson, C. W. and O'Hagan, A. (2005) Case studies in Gaussian process modelling of computer codes. In *Sensitivity Analysis of Model Output*, 476–485. Los Alamos National Laboratory.

Kennedy, M. C. and O'Hagan, A. (2001) Bayesian calibration of computer models (with discussion). *Journal of the Royal Statistical Society, B*, **63**, 425–464.

Kleijnen, J. P. C. (2008a) *Design and Analysis of Simulation Experiments*. New York: Springer.

— (2008b) Simulation experiments in practice: statistical design and regression analysis. *Journal of Simulation*, **2**, 19–27.

Lam, R. L. H., Welch, W. J. and Young, S. S. (2002) Uniform coverage designs for molecule selection. *Technometrics*, **44**, 99–109.

Lemieux, C. (2009) *Monte Carlo and Quasi-Monte Carlo Sampling*. New York: Springer.

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

Meyer, R. K. and Nachtsheim, C. J. (1995) The coordinate-exchange algorithm for constructing exact optimal designs. *Technometrics*, **37**, 60–69.

Nychka, D. (2005) *fields: Tools for spatial data.* URL `http://www.image.ucar.edu/GSP/Software/Fields`. R package version 3.04.

Overstall, A. M. and Woods, D. C. (2013) A strategy for Bayesian inference for computationally expensive models with application to the estimation of stem cell properties. *Biometrics*, in press.

Petelet, M., Iooss, B., Asserin, O. and Loredo, A. (2010) Latin hypercube sampling with inequality constraints. *Advances in Statistical Analysis*, **94**, 325–339.

Pronzato, L. and Müller, W. (2012) Design of computer experiments: space filling and beyond. *Statistics and Computing*, **22**, 681–701.

Qian, P. Z. G., Wu, H. and Wu, C. F. J. (2008) Gaussian process models for computer experiments with qualitative and quantitative factors,. *Technometrics*, **50**, 383–396.

Royle, J. A. (2002) Exchange algorithms for contructing large spatial designs. *Journal of Statistical Planning and Inference*, **100**, 121–134.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003) *The Design and Analysis of Computer Experiments*. New York: Springer.

SAS Proc Optex (1995) *SAS QC Software Vol. 1: Usage and Reference.* SAS Institute Inc., Cary NC.

Stinstra, E. D., den Hertog, D., Stehouwer, H. P. and Vestjens, A. (2003) Constrained maximin designs for computer experiments. *Technometrics*, **45**, 340–346.

Sykes, R. I., Parker, S. F., Henn, D. S., Cerasoli, C. P. and Santos, L. P. (1998) PC-SCIPUFF Version 1.2pd Technical Documentation. *Tech. Rep. ARAP 718*, ARAP Group, Princeton, NJ.

Wilks, D. S. (2006) *Statistical Methods in the Atmospheric Sciences.* Burlington, MA: Academic Press.

Wong, K. P. and Dong, Z. Y. (2005) Differential evolution, an alternative approach to evolutionary algorithm. In *Proceedings of the 13th International Conference on Intelligent Optimization and Control for Power Systems* (ed. K. Lee), 73–83. IEEE Publishing.

Zahran, A., Anderson-Cook, C. M. and Myers, R. H. (2003) Fracton of design space to assess the prediction capability of response surface designs. *Journal of Quality Technology*, **35**, 377–386.