Improving the Efficiency of Fully Bayesian Optimal Design of Experiments Using Randomised Quasi-Monte Carlo

Christopher C. Drovandi[*][‡] and Minh-Ngoc Tran[†][‡]

Abstract. Optimal experimental design is an important methodology for most efficiently allocating resources in an experiment to best achieve some goal. Bayesian experimental design considers the potential impact that various choices of the controllable variables have on the posterior distribution of the unknowns. Optimal Bayesian design involves maximising an expected utility function, which is an analytically intractable integral over the prior predictive distribution. These integrals are typically estimated via standard Monte Carlo methods. In this paper, we demonstrate that the use of randomised quasi-Monte Carlo can bring significant reductions to the variance of the estimated expected utility. This variance reduction can then lead to a more efficient optimisation algorithm for maximising the expected utility.

Keywords: approximate Bayesian computation, evidence, experimental design, importance sampling, mutual information, Laplace approximation, quasi-Monte Carlo.

1 Introduction

1.1 Background

A data collection process often involves variables that can be controlled. The field of optimal experimental design is devoted to developing methods to optimally choose values for these controllable variables so that the resulting data is likely to have a sufficient amount of information to address the aims of the analysis.

Fully Bayesian experimental design (see Bernardo and Smith (2000)) proceeds by specifying a utility function $U(\boldsymbol{d}, \boldsymbol{y}, \boldsymbol{\theta})$, which defines the worth of conducting an experiment that generated the observed data $\boldsymbol{y}$, assuming that the dataset was drawn from a model with parameter value $\boldsymbol{\theta}$ and the design $\boldsymbol{d}$ was applied. Of course, the experimental design $\boldsymbol{d}$ must be specified before data collection. A common approach in Bayesian design is to consider the expected utility over the prior predictive distribution of the data

$$U(\boldsymbol{d}) = \int_{\boldsymbol{y}} \int_{\boldsymbol{\theta}} U(\boldsymbol{d}, \boldsymbol{y}, \boldsymbol{\theta}) p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d}) p(\boldsymbol{\theta}) d\boldsymbol{\theta} d\boldsymbol{y}, \tag{1}$$

[*]School of Mathematical Sciences, Queensland University of Technology, Brisbane, 4000, Australia, c.drovandi@qut.edu.au

[†]The University of Sydney Business School, Sydney, 2006, Australia

[‡]ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS).

where $p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d})$ is the likelihood function and $p(\boldsymbol{\theta})$ is the prior distribution, which incorporates information from previous similar experiments and/or expert opinion. The optimal Bayesian design is the one that maximises the expected utility

$$\boldsymbol{d}^* = \arg \max_{\boldsymbol{d} \in \mathcal{D}} U(\boldsymbol{d}),$$

where $\mathcal{D}$ is the set of allowable designs. An alternative approach is when the utility is some functional of the posterior, in which $\boldsymbol{\theta}$ is integrated out producing the utility $U(\boldsymbol{d}, \boldsymbol{y})$. The expected utility is then

$$U(\boldsymbol{d}) = \int_{\boldsymbol{y}} U(\boldsymbol{d}, \boldsymbol{y}) p(\boldsymbol{y}|\boldsymbol{d}) d\boldsymbol{y}, \tag{2}$$

where $p(\boldsymbol{y}|\boldsymbol{d})$ is the prior predictive distribution of the data $\boldsymbol{y}$. An example is where $U(\boldsymbol{d}, \boldsymbol{y}, \boldsymbol{\theta}) = \log p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{d}) - \log p(\boldsymbol{\theta})$, which is the Shannon information gain (SIG). In this case, $U(\boldsymbol{d}, \boldsymbol{y})$ is the Kullback–Leibler divergence (KLD) between the prior and the posterior, $U(\boldsymbol{d}, \boldsymbol{y}) = \mathrm{KLD}(p(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{d}))$. It should be noted that under the SIG utility, the expected utility represents the mutual information (MI) between the parameter $\boldsymbol{\theta}$ and the potential future dataset $\boldsymbol{y}$ when the design $\boldsymbol{d}$ is applied. That is, the expected reduction in entropy about the parameter $\boldsymbol{\theta}$ brought about by the data $\boldsymbol{y}$. In general, the KLD is difficult to estimate precisely in a computationally efficient manner. A pragmatic alternative Bayesian parameter estimation utility is $U(\boldsymbol{d}, \boldsymbol{y}) = -\log(|\Sigma_{\boldsymbol{y}|\boldsymbol{d}}|)$ where $|\cdot|$ denotes the determinant of a matrix and $\Sigma_{\boldsymbol{y}|\boldsymbol{d}}$ is the posterior covariance matrix obtained when observing data $\boldsymbol{y}$ under design $\boldsymbol{d}$. Overstall et al. (2016) show that this utility is one component of the KLD when the prior and posterior are multivariate normal. In general, the posterior covariance matrix is easier to estimate than the KLD. It is important to note that other utility functions can be defined to reflect different goals such as prediction or model discrimination (see Section 3.4 for an example of the latter).

Whatever formulation is adopted for $U(\boldsymbol{d})$ (integration over $(\boldsymbol{y}, \boldsymbol{\theta})$ or just $\boldsymbol{y}$), the integral is generally analytically intractable. Our focus is on the formulation in (2) however the ideas in this paper can be adapted to the formulation in (1). A common approach in the Bayesian design literature is to estimate the integral via Monte Carlo (MC) integration

$$U_J(\boldsymbol{d}) = \frac{1}{J} \sum_{j=1}^{J} U(\boldsymbol{d}, \boldsymbol{y}_j),$$

assuming $(\boldsymbol{y}_j, \boldsymbol{\theta}_j) \overset{\mathrm{iid}}{\sim} p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d}) p(\boldsymbol{\theta})$ for $j = 1, \ldots, J$. That is, a parameter is first drawn from the prior, $p(\boldsymbol{\theta})$, then the dataset is drawn from the model, $p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d})$, conditional on this parameter value and the design $\boldsymbol{d}$. Each $U(\boldsymbol{d}, \boldsymbol{y}_j)$ calculation involves the computation or the approximation of the posterior distribution conditional on $\boldsymbol{y}_j$. Thus approximating a single expected utility calculation can be expensive, which subsequently needs to be maximised via some optimisation algorithm. The efficiency of the optimisation process will depend on the level of noise associated with the expected utility estimate. The computational challenges of the Bayesian optimal design problem are well summarised in Ryan et al. (2016).

1.2 Contribution and Outline

In this paper we propose the use of randomised quasi-Monte Carlo (RQMC) methods to improve the precision with which the expected utility is estimated for the same value of J. Any optimisation algorithm can then benefit from the reduction in the noise of expected utility approximations. It is well known that MC methods estimate expectations that converge at a rate of $\mathcal{O}(1/\sqrt{J})$ whereas RQMC can generally achieve a faster convergence rate (Owen, 1997).

The idea of QMC has been considered in pseudo-Bayesian design (see, for example, Bliemer et al. (2008)). Here the utility is some scalar function of the Fisher information matrix. In this case the utility is independent of the future data $\boldsymbol{y}$ and the expected utility is given by

$$U(\boldsymbol{d}) = \int_{\boldsymbol{\theta}} U(\boldsymbol{d}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta}.$$

This integral can be estimated by MC or RQMC by taking J samples of $\boldsymbol{\theta}$ from the prior in the relevant way. Our paper focuses on RQMC in the context of fully Bayesian design, which is much more difficult as it involves also the integral over the data space. We suspect that variance reduction is more critical in fully Bayesian design as it is much more computationally intensive than pseudo-Bayesian design. This is due to the fact that fully Bayesian design typically requires the repeated approximation of posterior quantities whereas pseudo-Bayesian design relies on an expression for the observed or Fisher information matrix only.

RQMC has recently received increasing attention in the statistics community. Gerber and Chopin (2015) show the efficiency of RQMC in sequential Monte Carlo, Tran et al. (2015) document a faster convergence in their variational Bayes updating procedure when the noisy gradient is computed using RQMC. In Bayesian decision problems, one selects the best decision by minimising the Bayes risk which is an integral of the loss function over the posterior (Robert, 2007, ch. 2) and it is important to estimate such integrals accurately. The RQMC method that we describe in this article can be applied to any Bayesian decision problem.

Some experimental design problems involve sequential decision making, where the loss function takes into account both the decision loss and the cost of conducting the experiment, and where the data are collected sequentially (Berger, 1985, ch. 7). The optimal stopping time and the best decision are selected to optimise an expected loss function (Berger, 1985, equation 7.2), which can be estimated accurately with RQMC. To reduce the complexity of the sequential design problem, a myopic approach may be adopted that only makes an optimal decision for the next observation only (e.g. Drovandi et al. (2013)). In this setting, the current 'prior' distribution may not be available in closed form and may be represented by a Monte Carlo sample. The application of RQMC is less clear in this setting. However, it would be possible to use importance sampling through a parametric approximation of the posterior, and the same RQMC idea in this paper can be used to estimate the expected utility of the design for the next observation conditional on the data obtained so far. In these respects, we conjecture

that RQMC can have a beneficial effect on sequential decision making problems, but a thorough investigation of this is beyond the scope of the current paper. In this article, we focus only on using RQMC for Bayesian experimental design problems where we know in advance how many observations we wish to design for and that this data will be collected in a single batch after the optimal design has been determined. This is often referred to in the literature as a static design.

In Section 2 we describe RQMC and how we implement it in the context of Bayesian design. Four examples that illustrate the benefits of RQMC are provided in Section 3. We conclude in Section 4.

2   Randomised Quasi-Monte Carlo for Bayesian Experimental Design

2.1   Randomised Quasi-Monte Carlo

Quasi-Monte Carlo

This section presents a short tutorial to QMC and in particular RQMC. A thorough treatment of the subject can be found in the monographs by Niederreiter (1992) and Dick and Pillichshammer (2010). Glasserman (2004, ch. 5) provides a more accessible introduction to the subject.

Consider the problem of estimating the following integral over the s-dimensional unit hypercube

$$I(f) = \int_{[0,1)^s} f(\boldsymbol{u})d\boldsymbol{u},$$

for some function f. Almost all Monte Carlo problems can be written in this form. It is convenient in QMC to take intervals to be closed on the left and open on the right.

Plain MC methods estimate this integral by an average of $f(\boldsymbol{u}_j)$ over J iid samples $\boldsymbol{u}_j \sim \mathcal{U}(0,1)^s$. It is well known that the convergence rate of such MC estimators is of order $J^{-1/2}$. QMC methods are a deterministic alternative that chooses the points $\boldsymbol{u}_j$ more evenly in $[0,1)^s$ than random in the sense that they minimize the so-called star-discrepancy of the point set. Consider a set of J points $P_J = \{\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{J-1}\}$ with $\boldsymbol{u}_j \in [0,1)^s$. Let $\mathcal{A}$ be a collection of subsets in $[0,1)^s$ of the form

$$A = \prod_{i=1}^{s}[0, a_i), \ \ a_i \in [0,1).$$

The star-discrepancy of the point set $P_J = \{\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{J-1}\}$ is defined as

$$D^*(P_J) = \sup_{A \in \mathcal{A}} \left| \frac{\#\{\boldsymbol{u}_j \in A\}}{J} - \text{Vol}(A) \right|,$$

where $\#\{\boldsymbol{u}_j \in A\}$ is the number of $\boldsymbol{u}_j$ belonging to A and $\text{Vol}(A)$ is the volume of set A. Clearly, $D^*(P_J)$ measures the non-uniformity of the point set P_J. The Koksma–Hlawka inequality states that

$$\left| I(f) - \frac{1}{J} \sum_{j=0}^{J-1} f(\boldsymbol{u}_j) \right| \le D^*(P_J)\sigma^2(f),$$

where $\sigma^2(f)$ is the variation of the function f. This inequality provides an upper bound on the approximation error and a criterion on searching for efficient QMC point sets P_J.

Several construction rules of low-discrepancy point sets P_J have been proposed, for which $D^*(P_J)$ is of order $J^{-1}(\log J)^{s-1}$. Before describing such construction rules, let us introduce the so-called van der Corput sequences (van der Corput, 1935). Given an integer $b \ge 2$, any integer $k \ge 0$ can be written as

$$k = \sum_{i=0}^{\infty} a_i(k)b^i, \;\; a_i(k) \in \{0, 1, \dots, b-1\},$$

with all, but finitely many, zero coefficients $a_i(k)$. Then the function

$$\psi_b(k) = \sum_{i=0}^{\infty} \frac{a_i(k)}{b^{i+1}}, \tag{3}$$

maps each integer $k \ge 0$ to a point in $[0, 1)$. The sequence $\{\psi_b(k), k = 0, 1, \dots\}$ is called the base-b van der Corput sequence. The table below gives the first 8 values of the base-2 van der Corput sequence. As k increases, the van der Corput sequence fills

k	0	1	2	3	4	5	6	7
$\psi_2(k)$	0	1/2	1/4	3/4	1/8	5/8	3/8	7/8

up the interval $[0, 1)$ in a very balanced way: they appear alternately on both sides of $1/2$, first $1/4$ and $3/4$, then $1/8$ and $5/8$, then $3/8$ and $7/8$ and so on. Because of this uniformity feature, van der Corput sequences are extensively used in the QMC literature for constructing low-discrepancy point sets.

The simplest construction of low-discrepancy point sets is the Halton–Hammersley sequence (Halton, 1960; Hammersley, 1960). Let $b_1, \dots, b_s$ be s positive integers such that their greatest common divisor is one. Then

$$\boldsymbol{u}_k = (\psi_{b_1}(k), \dots, \psi_{b_s}(k)), \;\; k = 0, 1, \dots$$

is a sequence of low-discrepancy points, which satisfies $D^*(\boldsymbol{u}_0, \dots, \boldsymbol{u}_{J-1}) = \mathcal{O}(J^{-1}(\log J)^s)$.

A more sophisticated rule is (t, m, s)-nets (Niederreiter, 1992). The set P_J of $J = b^m$ points, where $b \ge 2$ is an integer, is said to be a (t, m, s)-net in base b if for all $d_1, \dots, d_s \ge 0$ with $d_1 + \dots + d_s = m - t$, every elementary box of the form

$$\prod_{i=1}^{s} \left[\frac{a_i}{b^{d_i}}, \frac{a_i + 1}{b^{d_i}} \right), \;\; 0 \le a_i < b^{d_i} \text{ for } i = 1, \dots, s,$$

contains exactly b^t points of P_J. The volume of each elementary box is $1/b^{d_1+\dots+d_s} = 1/b^{m-t}$. A (t, m, s)-net has a high uniformity in the sense that the portion of points contained in each elementary box $b^t/J = 1/b^{m-t}$ is exactly the volume of the box.
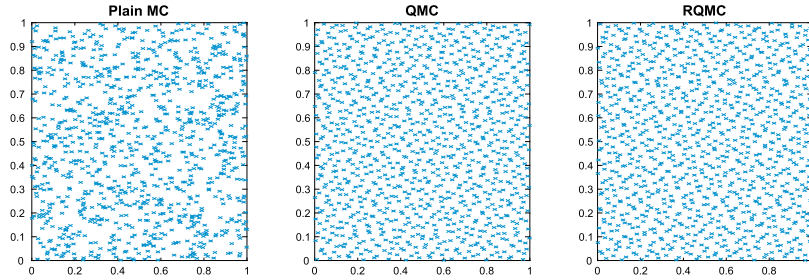
Figure 1: Plots of 1024 points generated by plain MC, (t, m, s)-net QMC and scrambled (t, m, s)-net RQMC with $t = 0$ and $s = 2$.

The reader is referred to Niederreiter (1992) and Dick and Pillichshammer (2010) for construction of (t, m, s)-nets.

Randomised Quasi-Monte Carlo

Unlike MC estimators which are stochastic, QMC estimators are deterministic. Applications in statistics often require randomness so that probability theory can be applied, for instance, for estimating approximation errors and constructing confidence intervals. Furthermore, we sometimes require an unbiased estimator of $I(f)$. To get the best of both MC and QMC, randomised versions of QMC have been proposed. The idea is that, by injecting a random element into a QMC sequence $P_J = \{\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{J-1}\}$, the resulting randomised QMC sequence $\tilde{P}_J = \{\tilde{\boldsymbol{u}}_0, \ldots, \tilde{\boldsymbol{u}}_{J-1}\}$ preserves the low-discrepancy property and, at the same time, $\tilde{\boldsymbol{u}}_j \sim \mathcal{U}[0, 1)^s$.

The simplest randomisation method is to shift the point set P_J by a random vector $\boldsymbol{r}$ uniformly distributed over $[0, 1)^s$

$$\tilde{\boldsymbol{u}}_j = (\boldsymbol{u}_j + \boldsymbol{r}) \bmod 1, \ \ j = 0, 1, \ldots, J - 1,$$

where $\alpha \bmod 1 = \alpha - \lfloor \alpha \rfloor$ with $\lfloor \alpha \rfloor$ the largest integer number that is smaller than α. Here the mod operator applies separately to each coordinate. It is easy to see that $\tilde{\boldsymbol{u}}_j$ is uniformly distributed over $[0, 1)^s$. Then $\widehat{I}(f) = (1/J) \sum f(\tilde{\boldsymbol{u}}_j)$ is an unbiased estimator of $I(f)$. It is important to note that the $\tilde{\boldsymbol{u}}_j$ are not independent of each other.

A more sophisticated RQMC method (Owen, 1997; Matousek, 1998) is to apply random permutations to the coefficients in the base-b expansion (3) of each number in a (t, m, s)-net. The resulting set is called a scrambled net. A visual comparison of plain MC, QMC and RQMC for $s = 2$ is shown in Figure 1. Owen (1997) shows that, given that f is smooth enough, the variance of $\widehat{I}(f)$ is of order $J^{-3}(\log J)^{s-1}$, compared to the order of J^{-1} for plain MC estimates. So given that the dimension s is not too large, RQMC estimates achieve a better convergence rate than plain MC estimates.

Here we describe the simplified version of Matousek (1998). Write P_J as a matrix of size $J \times s$ and consider a single element u_{ji} in this matrix with the base-b expansion

$u_{ji} = a_0/b + a_1/b^2 + \cdots, a_i \in \{0, 1, \ldots, b-1\}$. Let $\Pi^{(i)} = \{\pi_0^{(i)}, \pi_1^{(i)}, \ldots\}$ be independent random permutations of the set $\{0, 1, \ldots, b-1\}$. Then the scrambled version of u_{ji} is $\tilde{u}_{ji} = \pi_0^{(i)}(a_0)/b + \pi_1^{(i)}(a_1)/b^2 + \cdots$, i.e., we apply $\pi_\nu^{(i)}$ to the coefficients a_ν, $\nu = 0, 1, \ldots$. The same set of random permutations $\Pi^{(i)}$ is used for all elements in the ith column of P_J, different and independent sets $\Pi^{(i)}$ are used for different columns. The scrambled set $\tilde{P}_J = \{\tilde{\boldsymbol{u}}_0, \ldots, \tilde{\boldsymbol{u}}_{J-1}\}$ is a (t, m, s)-net with probability one and each $\tilde{\boldsymbol{u}}_j \sim U[0, 1)^s$. As the $f(\tilde{\boldsymbol{u}}_j)$ are dependent, it is in general harder to obtain central limit theorems (CLT) for the RQMC estimator $\widehat{I}(f) = (1/J) \sum f(\tilde{\boldsymbol{u}}_j)$. Loh (2003) obtains a CLT for RQMC estimators based on scrambled $(0, m, s)$-nets.

2.2 Estimating the Expected Utility

In the context of Bayesian design, simulation from the prior predictive, $\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d})p(\boldsymbol{\theta})$, can be written as a function (transformation) of uniform random variates. This involves a two stage process. In an abuse of notation we write the simulation of $\boldsymbol{\theta}$ from the prior as the function $\boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{u_\theta})$ where $\boldsymbol{u_\theta}$ is uniformly distributed over the hypercube $(0, 1)^{n_\theta}$ where n_θ is the dimension of $\boldsymbol{u_\theta}$. Then to produce a simulation of $\boldsymbol{y}$ we require an additional set of $n_{\boldsymbol{y}}$ uniform random numbers and we set $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{u_y}, \boldsymbol{\theta})$ where $\boldsymbol{u_y} \in (0, 1)^{n_y}$. We can write this more succinctly as $\boldsymbol{u} = (\boldsymbol{u_\theta}, \boldsymbol{u_y})^\top$ and set $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{u})$ where $\boldsymbol{u}$ is uniform over the hypercube $(0, 1)^{n_\theta + n_y}$. Then, we can re-write the integral in (2) as

$$U(\boldsymbol{d}) = \int_{\boldsymbol{u}} U(\boldsymbol{d}, \boldsymbol{y}(\boldsymbol{u}))d\boldsymbol{u}.$$

If we simulate $\boldsymbol{u}$ uniformly using pseudo random numbers then we recover the standard MC estimator in (2). However, if we generate $\boldsymbol{u}$ using the procedure outlined in the previous section then we obtain an estimator based on RQMC that should have lower variance.

We note that the choice of transformation function is not unique. For example, to simulate from a normal distribution, we could use the quantile function of the normal distribution (the inversion method) or the Box–Muller method (Box and Muller, 1958). In our examples in Section 3, we use the inversion method exclusively. Note that it is not necessarily guaranteed that the space filling properties of the RQMC numbers are preserved on the transformed space. Nonetheless, in Section 3, we obtain significant improvements.

We show later that increased precision afforded by the RQMC approach can improve the efficiency of optimisation algorithms for determining the design $\boldsymbol{d}^*$ that leads to the largest expected utility. There have been several methods developed for this optimisation task. We do not intend to give a complete overview here but give a few examples to provide a flavour. Müller et al. (2004) convert the optimisation into a Markov chain Monte Carlo (MCMC) simulation problem that simulates over the joint space of $(\boldsymbol{d}, \boldsymbol{y}, \boldsymbol{\theta})$ and admits a probability density proportional to the expected utility surface as the marginal in $\boldsymbol{d}$. Such an approach has been commonly used in the Bayesian design literature (e.g. Cook et al. (2008); Drovandi and Pettitt (2013)). Drovandi and Pettitt

---

Algorithm 1 Co-ordinate exchange design optimisation algorithm.

Input: Number of sweeps S, J, and a vector, $\boldsymbol{e}$, containing potential values for each component of the design vector $\boldsymbol{d}$.

Output: An approximation to the optimal design $\boldsymbol{d}^*$ and its corresponding estimated expected utility value.

1: Set initial design $\boldsymbol{d}^0$
2: Estimate expected utility $U^0 = U_J(\boldsymbol{d}^0)$
3: Set iteration counter $I \leftarrow 0$
4: **for** $s = 1$ **to** S **do**
5:     **for** $j = 1$ **to** $|\boldsymbol{e}|$ **do**
6:         Set $\boldsymbol{d}^+ \leftarrow \boldsymbol{d}^I$
7:         Replace jth element of $\boldsymbol{d}^+$, $d_j^+ = e_j$
8:         Calculate $U^+ = U_J(\boldsymbol{d}^+)$
9:         **if** $U^+ > U^I$ **then**
10:             Set $\boldsymbol{d}^I \leftarrow \boldsymbol{d}^+$
11:             Set $U^I \leftarrow U^+$
12:         **end if**
13:         Set $I \leftarrow I + 1$
14:     **end for**
15: **end for**

(2013) note some issues with this approach. In particular, the mixing of the MCMC can be poor and furthermore the optimal design is obtained by estimating the mode of the density proportional to the expected utility surface in a non-parametric fashionbased on only MCMC samples from this density. This approach does not scale with an increase in the length of $\boldsymbol{d}$.

Other approaches attempt to maximise $U(\boldsymbol{d})$ directly but their performance may be affected by the noise associated with $U_J(\boldsymbol{d})$. Huan and Marzouk (2014) implement various stochastic approximation algorithms to perform the optimisation. Gotwalt et al. (2009) apply the co-ordinate exchange (CE) algorithm (Meyer and Nachtsheim, 1995) in a pseudo-Bayesian design setting. The CE algorithm involves cycling through each of the design variables one-at-a-time, trialling a set of candidate replacements and choosing the best replacement if it improves the expected utility. Overstall and Woods (2016) extend this idea to an approximate CE algorithm that uses a Gaussian process emulator so that only a relatively small number of candidates are required. Weaver et al. (2016) consider a Bayesian optimisation algorithm that uses a Gaussian process emulator over the entire design space. Our main message is that all of these approaches can benefit from a reduction in the variance of $U_J(\boldsymbol{d})$. To illustrate the capability of RQMC in determining designs with higher expected utility than MC, we use the CE algorithm (shown in Algorithm 1). This algorithm requires the specification of a vector $\boldsymbol{e}$ containing potential values for each component of the design vector $\boldsymbol{d}$ and the number of sweeps, S, to perform over the design vector.

3   Examples

Here we consider four examples that illustrate the variance reduction in the estimate of the expected utility that can be achieved through RQMC. For a particular design $\boldsymbol{d}$ we obtain $U_J(\boldsymbol{d})$ under both MC and RQMC for several values of J. Note that for both MC and RQMC we use the same approach to compute/approximate $U(\boldsymbol{d}, \boldsymbol{y})$. We consider two different designs, one that is sub-optimal (e.g. a randomly selected design) and one that is closer to optimal. To obtain the closer-to-optimal design we use one run of the CE algorithm in conjunction with RQMC and $J = 100$. Note that we do not expend a great deal of effort into determining the most optimal design, we are simply interested in comparing MC and RQMC at two markedly different designs with different expected utilities. To estimate the precision of the estimated expected utility we obtain 50 (unless otherwise specified) independent values of $U_J(\boldsymbol{d})$ under each scenario, then compute the sample standard deviation to obtain $\mathrm{sd}(U_J(\boldsymbol{d}))$. We prefer the integration method that leads to lower values of the standard deviation. In this paper, we use the scrambled Sobol's net, i.e. the scrambled (t, m, s)-net in base $b = 2$.

We also investigate how much influence the reduction in variance provided by RQMC has in determining efficient designs. As mentioned above, we use the CE optimisation algorithm for this purpose. We run this for both RQMC and MC with different values of J. When a design is updated by the CE algorithm, we compute a gold standard value of its expected utility by using RQMC and $J = 1000$ and keep track of this gold standard value throughout the optimisation process. Since the CE algorithm is not a global optimisation method and that it may be impacted by the noise in the expected utility estimate, we run it several times from different random starting designs. For each individual run of the CE method, we use the same starting design for both MC and RQMC. We compare the MC and RQMC methods via the median value (over the repeated runs) of the gold standard estimate of the expected utility throughout the CE iterations.

For a fixed value of J, the only difference between run times of MC and RQMC for estimating $U_J(\boldsymbol{d})$ can be attributed to the time to generate random numbers for producing different values of $\boldsymbol{y}$. We find that there is only a very small additional cost for generating randomised QMC numbers over pseudo random numbers. The additional time required to estimate the expected utility via RQMC becomes negligible as the value of J and/or the time to approximate $U(\boldsymbol{d}, \boldsymbol{y})$ is increased.

3.1   Pharmacokinetics

We take this example from Ryan et al. (2014), which involves determining optimal blood sampling times for a pharmacokinetics (PK) model. PK studies involve the administration of a specified quantity of a drug to subjects and investigate the absorption, distribution and elimination of the drug and its metabolites. Let y_t be the observed concentration of the drug at time t. We assume that

$$y_t \sim \mathcal{N}(c(\boldsymbol{\theta})\mu_t(\boldsymbol{\theta}), \sigma^2 v_t(\boldsymbol{\theta})), \text{ where}$$
$$\mu_t(\boldsymbol{\theta}) = \exp(-\theta_1 t) - \exp(-\theta_2 t),$$

$$c(\boldsymbol{\theta}) = \frac{D}{\theta_3} \frac{\theta_2}{\theta_2 - \theta_1},$$

$$v_t(\boldsymbol{\theta}) = \left(1 + \frac{\tau^2}{\sigma^2} c(\boldsymbol{\theta})^2 \mu_t(\boldsymbol{\theta})^2\right),$$

where θ_1 is the elimination rate, θ_2 is the absorption rate and θ_3 is the 'the volume of distribution'. D is the administered dose, which is set fixed at 400 units. Additive and proportional error variances are set fixed at $\sigma^2 = 0.1$ and $\tau^2 = 0.01$, respectively. The goal of the experiment is the precise estimation of the parameter $\log(\boldsymbol{\theta}) = (\log(\theta_1), \log(\theta_2), \log(\theta_3))^\top$. The prior for θ_i is log-normal with a mean (on the log scale) of $\log(0.1)$, $\log(1)$ and $\log(20)$ for $i = 1, 2$ and 3, respectively. Each parameter has a variance (on the log scale) of 0.05 and the parameters are assumed independent *a priori*.

The design problem is to determine the optimal set of 15 sampling times $\boldsymbol{d} = (t_1, \ldots, t_{15})^\top$ where $t_1 < t_2 < \cdots < t_{15}$ and the sampling times are constrained to be at least 0.25 time units apart.

To estimate the posterior distribution we use a Laplace approximation (LA) of $\log(\boldsymbol{\theta})$. The LA is useful in optimal Bayesian design since the estimate of $U(\boldsymbol{d}, \boldsymbol{y})$ does not have noise associated with it (see Ryan et al. (2015) and Overstall et al. (2016)). Since the prior and posterior of $\log(\boldsymbol{\theta})$ are both normally distributed, there is an analytical expression for the KLD.

To apply RQMC we are required to write the simulation from the prior predictive distribution as a function of uniform random variates. The prior on each θ_i is log-normal so we can simulate it via $\theta_i = \exp(\mu_i + \sigma_i \Phi^{-1}(u_{\boldsymbol{\theta}}^i))$ where $\Phi^{-1}(\cdot)$ is the quantile function of a standard normal random variate, $u_{\boldsymbol{\theta}}^i \sim \mathcal{U}(0, 1)$, and μ_i and σ_i are the prior mean and standard deviation of $\log \theta_i$. After simulation from the prior to obtain $\boldsymbol{\theta}$, we can easily simulate the response as it is normally distributed, $y_{t_k} = c(\boldsymbol{\theta})\mu_{t_k}(\boldsymbol{\theta}) + \sigma\sqrt{v_{t_k}(\boldsymbol{\theta})}\Phi^{-1}(u_{\boldsymbol{y}}^k)$ where $u_{\boldsymbol{y}}^k \sim \mathcal{U}(0, 1)$ for $k = 1, \ldots, 15$. Thus we can write the expected utility with respect to the prior predictive distribution as an integral over the $3 + 15 = 18$ dimensional unit hypercube.

The sub-optimal design we consider is the evenly spaced design $\boldsymbol{d} = (1, 2, \ldots, 15)^\top$. The closer-to-optimal design is $\boldsymbol{d} = (0.75, 1.0, 1.25, 1.75, 3.75, 4.0, 4.75, 5.0, 5.25, 10.5, 11.0, 11.25, 13.0, 15.0, 15.75)^\top$. Based on $J = 1000$ and RQMC the expected utilities of these designs are 7.02 and 7.37, respectively. The results are shown in Figure 2. It is evident that RQMC clearly outperforms MC for any value of J.

Figure 3 shows the median of the gold standard expected utility when performing 50 independent runs of the CE algorithm for both MC and RQMC for a variety of J values. Here we set $\boldsymbol{e} = (0.25, 0.5, \ldots, 24)^\top$ and $S = 5$. It is clear that the RQMC method is beneficial in obtaining designs with higher expected utility.
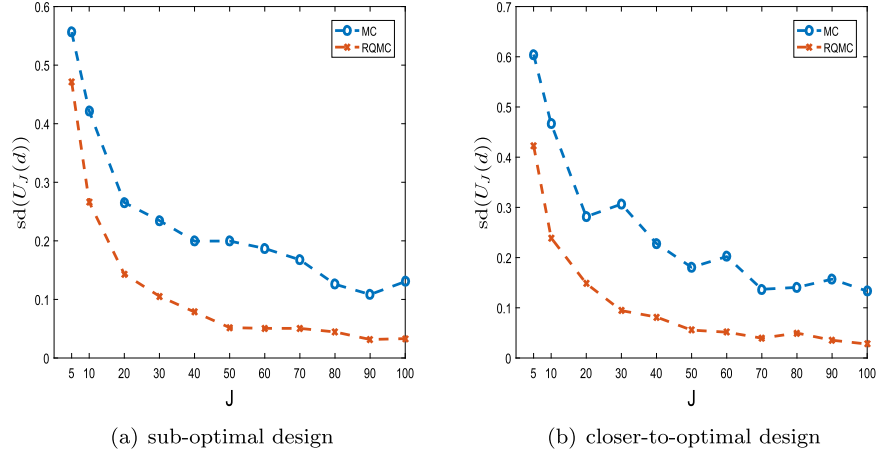
(a) sub-optimal design   (b) closer-to-optimal design

Figure 2: Estimated standard deviations of the estimated expected utilities $U_J(\boldsymbol{d})$ for different values of J for the PK example. Subfigure (a) shows results for a sub-optimal evenly spaced design whilst subfigure (b) shows results for a closer-to-optimal design. Results based on MC integration are shown as circles whilst results based on RQMC are shown as crosses. Points are connected via linear interpolation.

3.2 Logistic Regression

We consider the logistic regression example of Overstall and Woods (2016). The response is binary, $y_i \sim \mathcal{B}(p_i)$, where

$$\text{logit}(p_i) = \beta_0 + \sum_{j=1}^{4} \beta_j x_{ij}, \text{ for } i = 1, \ldots, n,$$

where n is the number of observations. The model parameter is $\boldsymbol{\theta} = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4)^\top$. The observed vector of responses is denoted as $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ and the design vector is the concatenation of the controllable elements of the design matrix, $\boldsymbol{d} = \{x_{i,j}; i = 1, \ldots, n, j = 1, \ldots, 4\}$ and is of length $n \times 4$. The design space has $x_{ij} \in (-1, 1)$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, 4$. We design for $n = 24$ independent observations. The prior distribution allocated by Overstall et al. (2016) is $\beta_0 \sim \mathcal{N}(0,3)$, $\beta_1 \sim \mathcal{N}(7,3)$, $\beta_2 \sim \mathcal{N}(8,3)$, $\beta_3 \sim \mathcal{N}(-3,3)$, $\beta_4 \sim \mathcal{N}(0.5,3)$, and that all parameters are independent *a priori*.

We consider the MI utility for $U(\boldsymbol{d})$. In this case $U(\boldsymbol{d}, \boldsymbol{y})$ is the KLD between the prior and the posterior, and may be written as

$$U(\boldsymbol{d}, \boldsymbol{y}) = \int_{\boldsymbol{\theta}} \log(p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d}))p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{d})d\boldsymbol{\theta} - \int_{\boldsymbol{\theta}} p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{d})p(\boldsymbol{\theta})d\boldsymbol{\theta}.$$

We consider an MC approximation of the above utility based on importance sampling (IS) where the importance distribution is simply the prior. Denote a large collection of
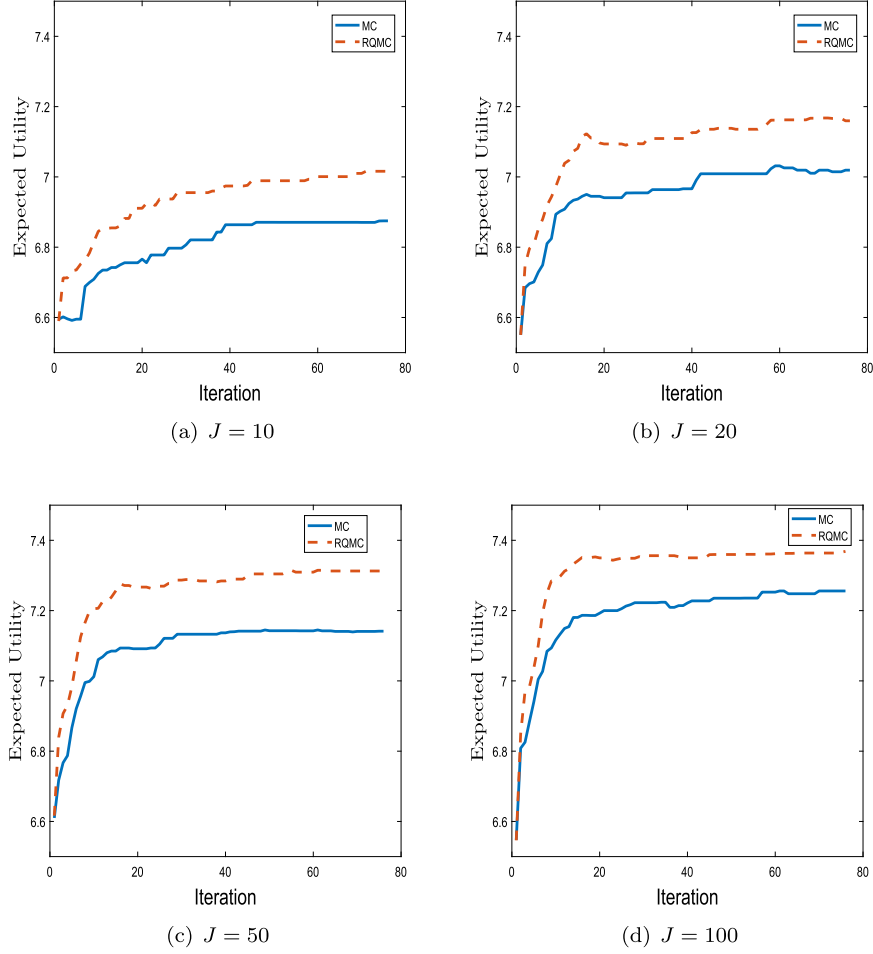
Figure 3: The median of the expected utility for the PK example over 50 independent runs of the CE optimisation algorithm with different starting designs. Results for various values of J used in the optimisation process are considered. Results are shown when using MC (solid) and RQMC (dashed) for estimating the expected utility during the CE algorithm.

N samples taken from the prior as $\Theta = \{\boldsymbol{\theta}_i\}_{i=1}^N \stackrel{\text{iid}}{\sim} p(\boldsymbol{\theta})$. Here we set $N = 100,000$. A weight is assigned to each particle, $W_i \propto p(\boldsymbol{y}|\boldsymbol{\theta}_i, \boldsymbol{d})$ such that $\sum_{i=1}^N W_i = 1$, so that $\{W_i, \boldsymbol{\theta}_i\}_{i=1}^N$ forms a particle approximation of $p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{d})$. Then the approximation of $U(\boldsymbol{d}, \boldsymbol{y})$ is given by

$$U_N(\boldsymbol{d}, \boldsymbol{y}) = \sum_{i=1}^N W_i \log(p(\boldsymbol{y}|\boldsymbol{\theta}_i, \boldsymbol{d})) - \frac{1}{N} \sum_{i=1}^N p(\boldsymbol{y}|\boldsymbol{\theta}_i, \boldsymbol{d}).$$

1	1	-1	-1		-1.0	0.0	-1.0	0.5
-1	1	-1	1		-1.0	1.0	1.0	1.0
-1	-1	1	-1		1.0	-1.0	1.0	1.0
1	-1	1	1		1.0	-1.0	1.0	1.0
-1	-1	-1	-1		-1.0	1.0	-1.0	-1.0
1	1	1	1		1.0	-1.0	-1.0	-1.0
-1	-1	-1	-1		-1.0	1.0	-1.0	0.0
1	-1	1	1		-1.0	0.0	1.0	1.0
-1	1	-1	-1		-1.0	-1.0	1.0	1.0
1	-1	-1	1		0.0	-1.0	-1.0	1.0
1	-1	-1	-1		1.0	-1.0	-1.0	-1.0
-1	-1	1	-1		0.5	-1.0	-1.0	-1.0
1	1	-1	-1		1.0	-1.0	-1.0	1.0
1	1	1	-1		-1.0	1.0	1.0	1.0
-1	1	1	-1		-1.0	1.0	1.0	-1.0
-1	-1	1	1		0.5	0.0	1.0	1.0
-1	-1	1	-1		1.0	-1.0	-1.0	1.0
1	1	-1	1		-1.0	1.0	1.0	-1.0
1	-1	1	1		1.0	-1.0	-1.0	1.0
1	-1	1	1		-1.0	0.5	1.0	-1.0
1	-1	-1	-1		-1.0	1.0	1.0	-1.0
-1	1	1	-1		1.0	-1.0	1.0	1.0
1	-1	-1	-1		1.0	-1.0	1.0	-1.0
-1	-1	-1	-1		-1.0	0.5	-1.0	1.0

Table 1: Designs used in the logistic regression example. Shown is a sub-optimal (random) design (left) and a closer-to-optimal design (right).

The advantage of using IS in the context of Bayesian design is that the same collection Θ can be used for each dataset $\boldsymbol{y}$ drawn during the design optimisation process. However, the optimal design obtained will be dependent on Θ, which we denote as $\boldsymbol{d}_\Theta^*$. To eliminate the dependence on Θ it would be necessary to re-generate a new collection of Θ for each $\boldsymbol{y}$. However, this leads to an increase in the computation time. We consider both cases here. We refer to the former as 'fixed Θ' and the latter as 'new Θ'. Here we perform 100 independent repeats to estimate $\mathrm{sd}(U_J(\boldsymbol{d}))$.

To compare the performance of MC and RQMC, we consider two values of $\boldsymbol{d}$ (see Table 1). The first is a random design where each element x_{ij} is equal to -1 or 1 with equal probability. Using $J = 1000$, RQMC and new Θ, the expected utilities for the random and closer-to-optimal designs are 2.94 and 4.29, respectively. Intuitively, the utility, $U(\boldsymbol{d}, \boldsymbol{y})$, may be more difficult to estimate at efficient designs as they provide more information about the parameters thus leading to a lower effective sample size in the IS approximation. This will naturally lead to more variability associated with $U_N(\boldsymbol{d}, \boldsymbol{y})$.

Since the prior on each parameter is normal we can simulate from it using the quantile function of the normal distribution (similar to the previous example). The
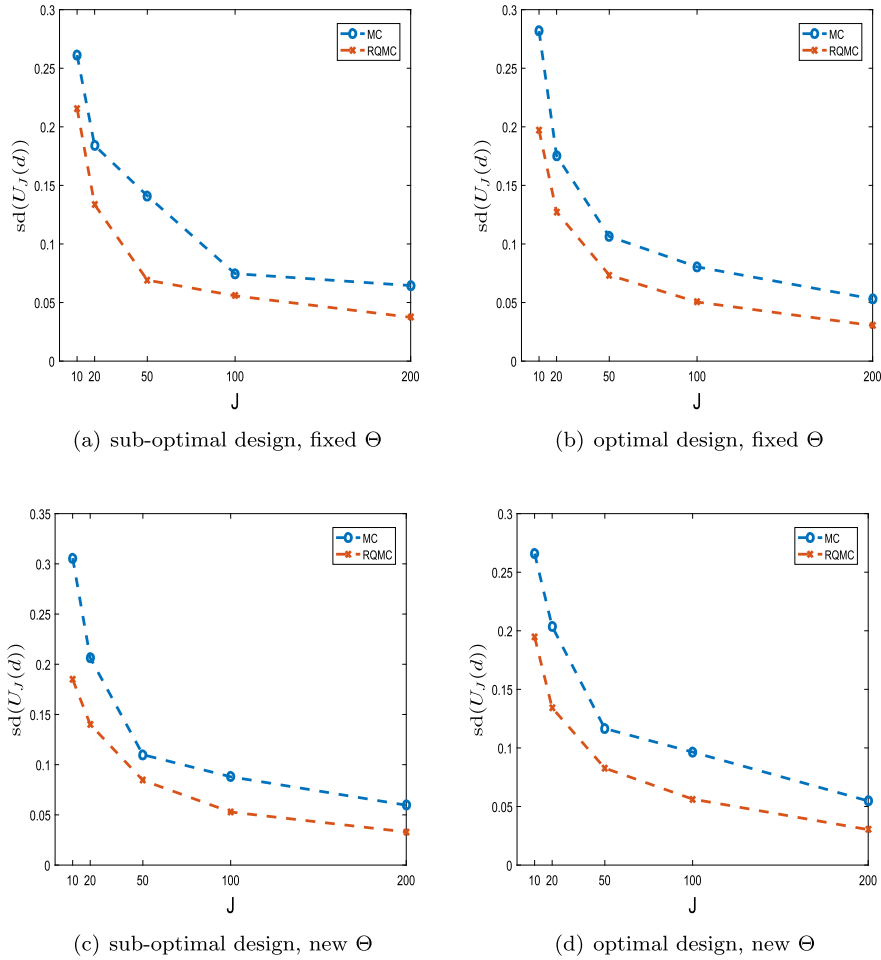
(a) sub-optimal design, fixed Θ (b) optimal design, fixed Θ

(c) sub-optimal design, new Θ (d) optimal design, new Θ

Figure 4: Estimated standard deviations of the estimated expected utilities $U_J(\boldsymbol{d})$ for different values of J for the logistic regression example. Subfigures (a) and (c) show results for a sub-optimal random design whilst subfigures (b) and (d) show results for a more optimal design. Subfigures (a) and (b) use the same importance samples to estimate $U(\boldsymbol{d}, \boldsymbol{y})$ for different $\boldsymbol{y}$ whilst subfigures (c) and (d) generate new importance samples for each $\boldsymbol{y}$. Results based on MC integration are shown as circles whilst results based on RQMC are shown as crosses. Points are connected via linear interpolation.

response is binary so we may simulate it as $y_i = \mathbb{I}(u_{\boldsymbol{y}}^i < p_i)$ for $i = 1, \ldots, n$ where p_i is a function of the model parameter, $u_{\boldsymbol{y}}^i \sim \mathcal{U}(0,1)$ and $\mathbb{I}(\cdot)$ is the indicator function.

The comparison of the standard deviation of the estimated expected utility is shown in Figure 4. The top row has fixed Θ while the bottom row has new Θ. The left column

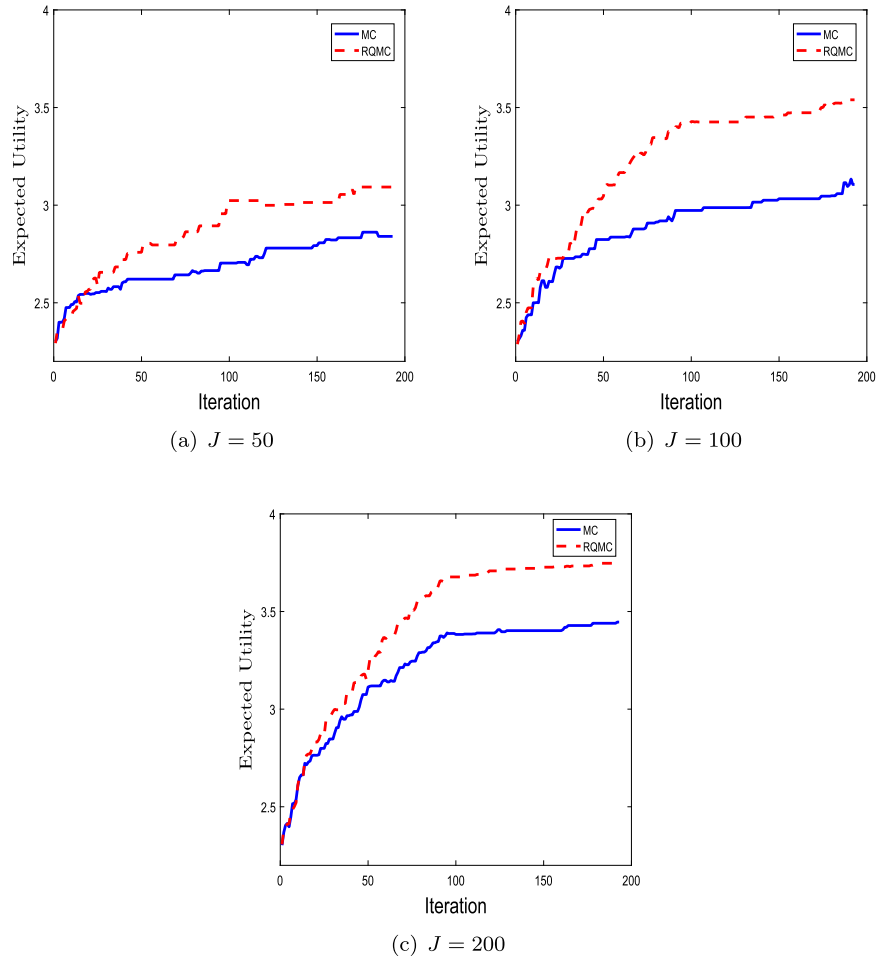(a) $J = 50$

(b) $J = 100$

(c) $J = 200$

Figure 5: The median of the expected utility for the logistic regression example over 30 independent runs of the CE optimisation algorithm with different starting designs. Results for various values of J used in the optimisation process are considered. Results are shown when using MC (solid) and RQMC (dashed) for estimating the expected utility during the CE algorithm.

is based on the sub-optimal design whereas the right column uses the closer-to-optimal design. In all four scenarios and for the majority of J values the RQMC approach brings a considerable variance reduction.

Figure 5 shows the median of the gold standard expected utility when performing 30 independent runs of the CE algorithm for both MC and RQMC for a variety of J values. Here we set $\boldsymbol{e} = (-1, -0.5, 0, 0.5, 1)^{\top}$ and $S = 2$. Again it is evident that RQMC provides a better chance of finding designs with higher expected utility than MC.

3.3   Susceptible-Infected Example

We investigate an example involving a stochastic infectious disease model considered by Cook et al. (2008); Drovandi and Pettitt (2013). Such models are important for gaining an understanding of how a disease is transmitted and for assessing the impact that various intervention strategies may have on the spread of the disease. Scientists may be interested in the optimal times to observe an epidemic in order to increase the chance of learning about the parameter of an infectious disease model or to learn the structure of the model itself, or both. In this section we consider the goal of parameter estimation while in the next section the aim is model discrimination.

The model of interest in this section is the susceptible-infected (SI) model, which is a continuous time Markov chain (or Markov process). We denote the number of susceptibles at time t as $S(t)$. We have that $P(S(t + \Delta_t) = i - 1|S(t) = i) = (b_1 + b_2(n - i))i\Delta_t + o(\Delta_t)$ where Δ_t is an infinitesimal time such that at most one event can occur during that time. The parameter b_1 represents the natural death rate of susceptibles while b_2 is the transmission rate of the disease. We are interested in the times to observe the process (here four times) to gain the most information about the parameter, $\boldsymbol{\theta} = (b_1, b_2)^\top$. The parameters are independent *a priori* and log-normally distributed: $b_1 \sim \mathcal{LN}(-3.6, 0.1024)$ and $b_2 \sim \mathcal{LN}(-4.5, 0.16)$.

We assume a closed population of size 50 and all individuals are susceptible at time 0. Further, we assume that it is only possible to observe the state of the system at various discrete times, so that the process is only partially observed. The likelihood function for Markov processes involves the computation of the transition probability matrix for moving between states of the Markov chain (here the number of infected individuals) during some time interval $t_a - t_b$ where $b > a$. For continuous time Markov chains, the transition probability matrix is calculated via the matrix exponential (see Moler and van Loan (2003)), which is given by $\exp(G(t_j - t_i))$, where G is the so-called generator matrix which contains as its (i, j)th element the transition intensity of moving from state i to state j if $i \neq j$. Each diagonal element contains the negative sum of its corresponding row. The matrix exponential is only computationally feasible for very small populations. Further, the number of likelihood calculations required during the optimal design process may render this approach infeasible for many Markov processes of interest. This motivated Drovandi and Pettitt (2013) to develop a likelihood-free approach to experimental design using approximate Bayesian computation (ABC, see also Hainy et al. (2014)). Drovandi and Pettitt (2013) use ABC rejection to generate samples from an approximate posterior, which is used to estimate the utility $U(\boldsymbol{d}, \boldsymbol{y}) = -\log(|\Sigma_{\boldsymbol{y}|\boldsymbol{d}}|)$ using the sample covariance matrix. ABC rejection involves generating and storing a set of N simulations from the prior predictive distribution, which we denote as $(\Theta, \mathcal{Y})$. Then, a small proportion, α, of these simulations are kept that are 'closest' to the dataset $\boldsymbol{y}$. The ABC approach of Drovandi and Pettitt (2013) for estimating the utility $U(\boldsymbol{d}, \boldsymbol{y})$ is shown in Algorithm 2. We use the same discrepancy function as in Drovandi and Pettitt (2013):

$$\rho(\boldsymbol{y}, \boldsymbol{x}|\boldsymbol{d}) = \sum_{i=1}^{n} \frac{|y_i - x_i|}{\text{std}(x_i|d_i)}, \tag{4}$$

where n is the number of observations to collect. The value $\text{std}(x_i|d_i)$ is the empirical prior predictive standard deviation of the data at time point d_i.

Algorithm 2 ABC rejection algorithm used in Drovandi and Pettitt (2013).

Input: Design, $\boldsymbol{d}$, a potential future dataset, $\boldsymbol{y}$, prior distribution, $p(\boldsymbol{\theta})$, number of prior predictive simulations, N, discrepancy function, $\rho(\cdot|\boldsymbol{d})$ and proportion of parameter samples to keep, α, to estimate the ABC posterior distribution

Output: An ABC estimate of $U(\boldsymbol{d}, \boldsymbol{y}) = -\log(|\Sigma_{\boldsymbol{y}|\boldsymbol{d}}|)$

1: Generate $\boldsymbol{\theta}^i \sim p(\boldsymbol{\theta})$ for $i = 1, \ldots, N$
2: Simulate $\boldsymbol{x}^i \sim p(\boldsymbol{y}|\boldsymbol{\theta}^i, \boldsymbol{d})$ for $i = 1, \ldots, N$
(Note that steps 1 and 2 can be performed prior to the optimal design process and the simulated data can be stored at a discretised grid of the design space)
3: Compute discrepancies $\rho^i = \rho(\boldsymbol{y}, \boldsymbol{x}^i|\boldsymbol{d})$ for $i = 1, \ldots, N$, creating particles $\{\boldsymbol{\theta}^i, \rho^i\}_{i=1}^N$

4: Sort the particle set via the discrepancy ρ such that $\rho^1 \leq \rho^2 \leq \cdots \leq \rho^N$
5: Calculate $\epsilon = \rho^{\lfloor \alpha N \rfloor}$ (where $\lfloor \cdot \rfloor$ denotes the floor function). The ABC posterior samples consist of the set $\{\boldsymbol{\theta}^i|\rho^i \leq \epsilon\}_{i=1}^N$
6: Estimate $\Sigma_{\boldsymbol{y}|\boldsymbol{d}}$ via the sample covariance matrix based on $\{\boldsymbol{\theta}^i|\rho^i \leq \epsilon\}_{i=1}^N$ and compute $U(\boldsymbol{d}, \boldsymbol{y}) = -\log(|\Sigma_{\boldsymbol{y}|\boldsymbol{d}}|)$

The advantage of ABC rejection, as with IS in the previous example, is that the same simulations, $(\Theta, \mathcal{Y})$, can be re-used for each $\boldsymbol{y}$ generated during the optimal design process. We refer to this as 'fixed $(\Theta, \mathcal{Y})$'. We also implement the more computationally demanding process of generating a new $(\Theta, \mathcal{Y})$ for each new $\boldsymbol{y}$, which we refer to as 'new $(\Theta, \mathcal{Y})$'. The ABC rejection approach requires a discretisation of the design space, whichhere is the sampling times between 0 and 10 days. As in Drovandi and Pettitt (2013), we use a design space with a time increment of 0.25 days. For the sub-optimal design we set $\boldsymbol{d} = (1, 2, 3, 4)^\top$. This design is chosen arbitrarily to be less efficient than the closer-to-optimal design. The closer-to-optimal design is $\boldsymbol{d} = (1.5, 3.75, 5.25, 9.75)^\top$. Using $J = 1000$, RQMC and new $(\Theta, \mathcal{Y})$, we estimate the expected utilities of these designs to be 20.89 and 21.76, respectively. The prior value of the utility is roughly 19.72.

The priors are log-normal so that prior simulation can proceed in the same way as the PK example. To simulate from the model given a value of $\boldsymbol{\theta}$ we can use the algorithm of Gillespie (1977), where the time between a loss of a susceptible has an exponential distribution with a rate parameter that depends on $\boldsymbol{\theta}$ and the current number of susceptibles in the simulation, which we write as $\eta(\boldsymbol{\theta}, S(t)) = (b_1 + b_2(50 - S(t)))S(t)$ where t is the current time of the simulation. To produce the simulation, the value of $S(t)$ is recorded at sampling times $\boldsymbol{d}$. Since there are 50 susceptibles at time 0 the maximum number of random numbers required to simulate the data is 50. We can simulate the time between each loss of susceptible using the quantile function of the exponential distribution, $t_j - t_{j-1} = -\log(1 - u_{\boldsymbol{\theta}}^j)/\eta(\boldsymbol{\theta}, S(t_{j-1}))$ where $u_{\boldsymbol{\theta}}^j \sim \mathcal{U}(0, 1)$ for $j = 1, \ldots, 50$. Most simulations will not require the use of all 50 uniform random numbers, but we store all of the potential randomised Sobol's numbers nonetheless.

The results for the four different scenarios for different values of J are shown in Figure 6. Again, RQMC is clearly preferred to MC.
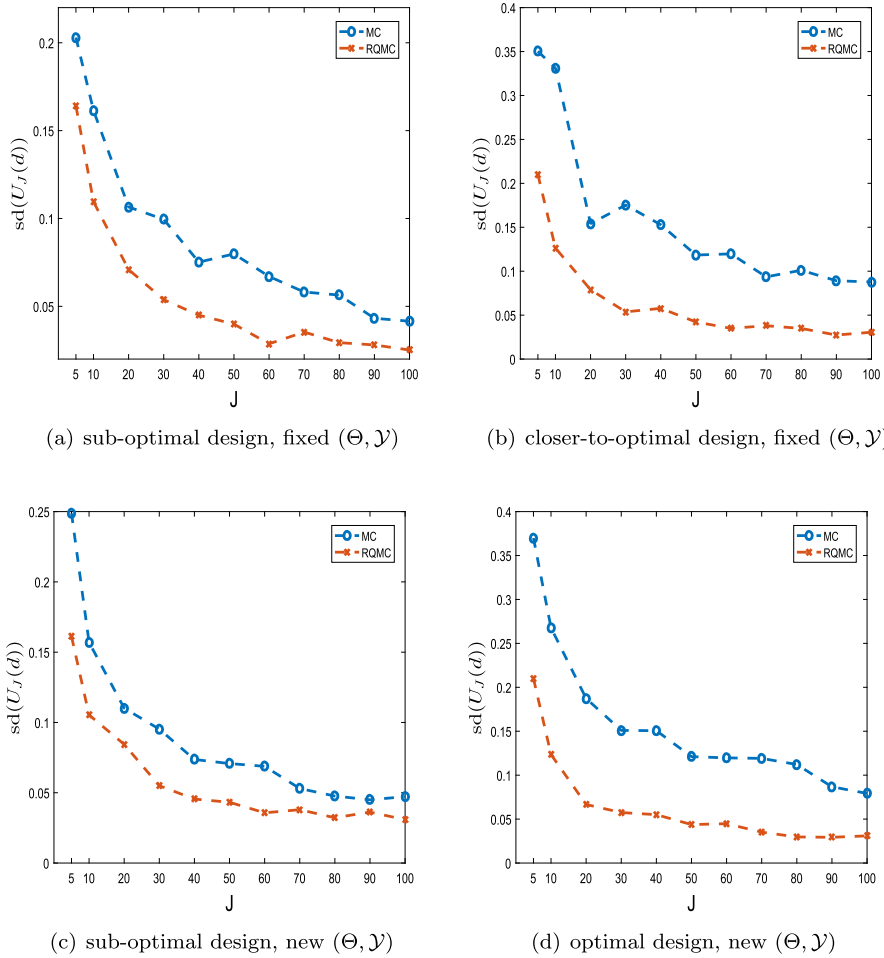
(a) sub-optimal design, fixed $(\Theta, \mathcal{Y})$

(b) closer-to-optimal design, fixed $(\Theta, \mathcal{Y})$

(c) sub-optimal design, new $(\Theta, \mathcal{Y})$

(d) optimal design, new $(\Theta, \mathcal{Y})$

Figure 6: Estimated standard deviations of the estimated expected utilities $U_J(\boldsymbol{d})$ for different values of J for the SI example. Subfigures (a) and (c) show results for a sub-optimal design whilst subfigures (b) and (d) show results for a more optimal design. Subfigures (a) and (b) use the same importance samples to estimate $U(\boldsymbol{d}, \boldsymbol{y})$ for different $\boldsymbol{y}$ whilst subfigures (c) and (d) generate new importance samples for each $\boldsymbol{y}$. Results based on MC integration are shown as circles whilst results based on RQMC are shown as crosses. Points are connected via linear interpolation.

Figure 7 shows the median of the gold standard expected utility when performing 50 independent runs of the CE algorithm for both MC and RQMC for a variety of J values. Here we set $\boldsymbol{e} = (0.25, 0.5, \ldots, 10)^\top$ and $S = 5$. In this example it appears that RQMC is only beneficial relative to MC for small J. In this case, the design vector is only of length four and thus it is not difficult to find a design with relatively high expected utility.
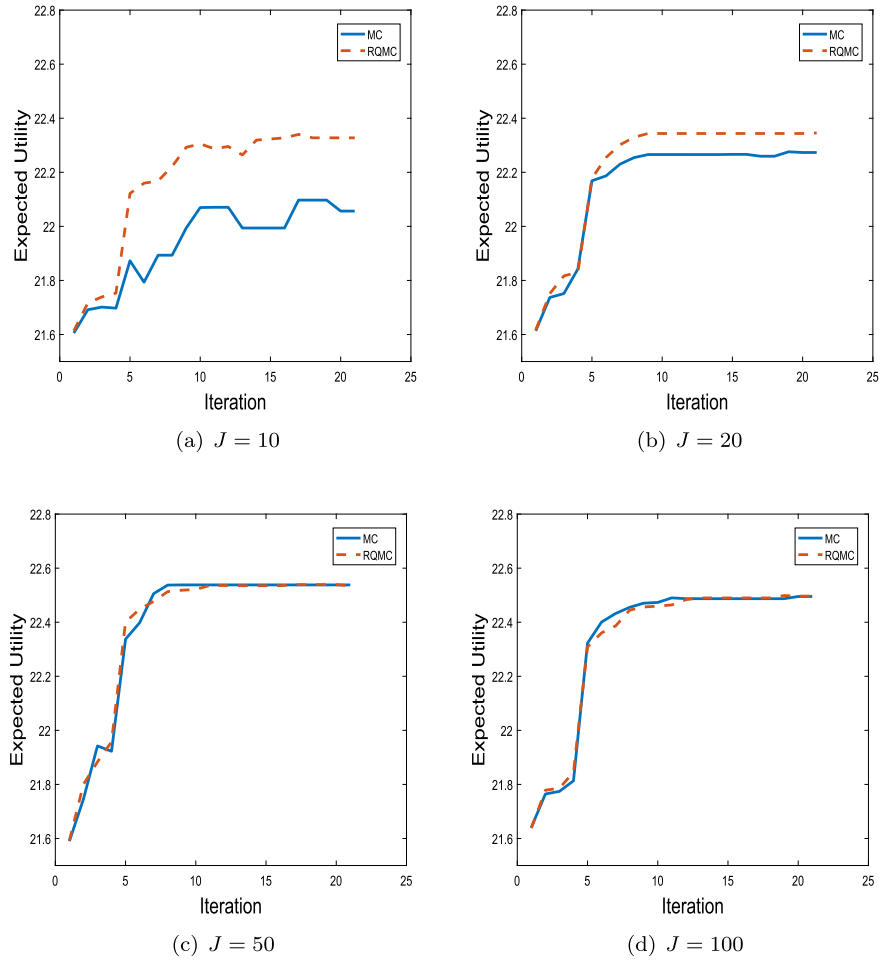
Figure 7: The median of the expected utility for the SI example over 50 independent runs of the CE optimisation algorithm with different starting designs. Results for various values of J used in the optimisation process are considered. Results are shown when using MC (solid) and RQMC (dashed) for estimating the expected utility during the CE algorithm.

3.4   Model Discrimination Example

In many scenarios there may be uncertainty about the true underlying model. We consider the M-closed perspective of Bernardo and Smith (2000), which assumes that the true data generating process is unknown but is one of K candidate models described by the random variable $M \in \{1, 2, \ldots, K\}$. For model $M = m$, we denote its prior probability as $p(m)$, the prior for its model parameter as $p(\boldsymbol{\theta}_m | m)$ and the likelihood function under that model as $p(\boldsymbol{y} | m, \boldsymbol{\theta}_m, \boldsymbol{d})$. The expected utility to accommodate this

model uncertainty can be written as

$$U(\boldsymbol{d}) = \sum_{m=1}^{K} p(m) \int_{\boldsymbol{y}} U(\boldsymbol{d}, \boldsymbol{y}, m) p(\boldsymbol{y}|m, \boldsymbol{d}) d\boldsymbol{y},$$

which can be estimated as

$$U_J(\boldsymbol{d}) = \sum_{m=1}^{K} p(m) \frac{1}{J} \sum_{j=1}^{J} U(\boldsymbol{d}, \boldsymbol{y}_j, m),$$

where $\boldsymbol{y}_j \sim p(\boldsymbol{y}|m, \boldsymbol{d})$ for $j = 1, \ldots, J$ and $m = 1, \ldots, K$. If determining the design to best discriminate between models is of interest, then one possibility is to set $U(\boldsymbol{d}, \boldsymbol{y}, m) = \log p(m|\boldsymbol{y}, \boldsymbol{d})$ where $p(m|\boldsymbol{y}, \boldsymbol{d})$ is the posterior probability of model m (see, for example, Box and Hill (1967); Drovandi et al. (2014)). This choice amounts to maximising the mutual information between the model indicator M and the potential future dataset $\boldsymbol{y}$.

Here we consider the example specified in Dehideniya et al. (2016), which involves discriminating between the SI model (above) and also the 'death' model, which has $P(S(t + \Delta_t) = i - 1|S(t) = i) = b_1 \Delta_t + o(\Delta_t)$. We use a parameter prior of $b_1 \sim \mathcal{LN}(-0.48, 0.15^2)$ for the death model and $b_1 \sim \mathcal{LN}(-1.1, 0.2^2)$, $b_2 \sim \mathcal{LN}(-4.5, 0.6^2)$ for the SI model. The two models are assumed equally likely *a priori*. To estimate the utility for some generated dataset $\boldsymbol{y}$, the ABC rejection algorithm is used. Out of all the prior predictive simulations that are within some distance of the dataset $\boldsymbol{y}$, the proportion of those belonging to model m is used to estimate the posterior model probability of model m (see Dehideniya et al. (2016) and the references therein for more details).

To utilise RQMC here a very similar procedure is required as in the previous example. The only difference is that two sets of J prior predictive simulations are required, one for each of the two models. Here we perform 100 independent repeats to estimate $\text{sd}(U_J(\boldsymbol{d}))$. The sub-optimal design is $\boldsymbol{d} = (5, 10)$ days (chosen arbitrarily to be less efficient than the closer-to-optimal design) and the closer-to-optimal design is $\boldsymbol{d} = (0.75, 4.75)$ days. The expected utilities of these designs based on RQMC and $J = 1000$ are -0.64 and -0.44, respectively. The results for both of these designs and also whether or not a new set of prior predictive simulations is used for the ABC rejection for each value of j is shown in Figure 8. Despite the MC error associated with the $U(\boldsymbol{d}, \boldsymbol{y}, m)$ calculation, a significant gain is achieved with RQMC without any additional computational cost. However, owing to the low dimensional design space, RQMC does not speed up the rate of convergence to designs with high expected utility.

4   Conclusion

This paper has demonstrated that RQMC can lead to substantial variance reduction in expected utility estimates for fully Bayesian experimental design. Our focus was on applying RQMC to increase the precision of $U_J(\boldsymbol{d})$ for fixed J. Our results suggest

(a) sub-optimal design, fixed $(\Theta, \mathcal{Y})$ (b) optimal design, fixed $(\Theta, \mathcal{Y})$

(c) sub-optimal design, new $(\Theta, \mathcal{Y})$ (d) optimal design, new $(\Theta, \mathcal{Y})$
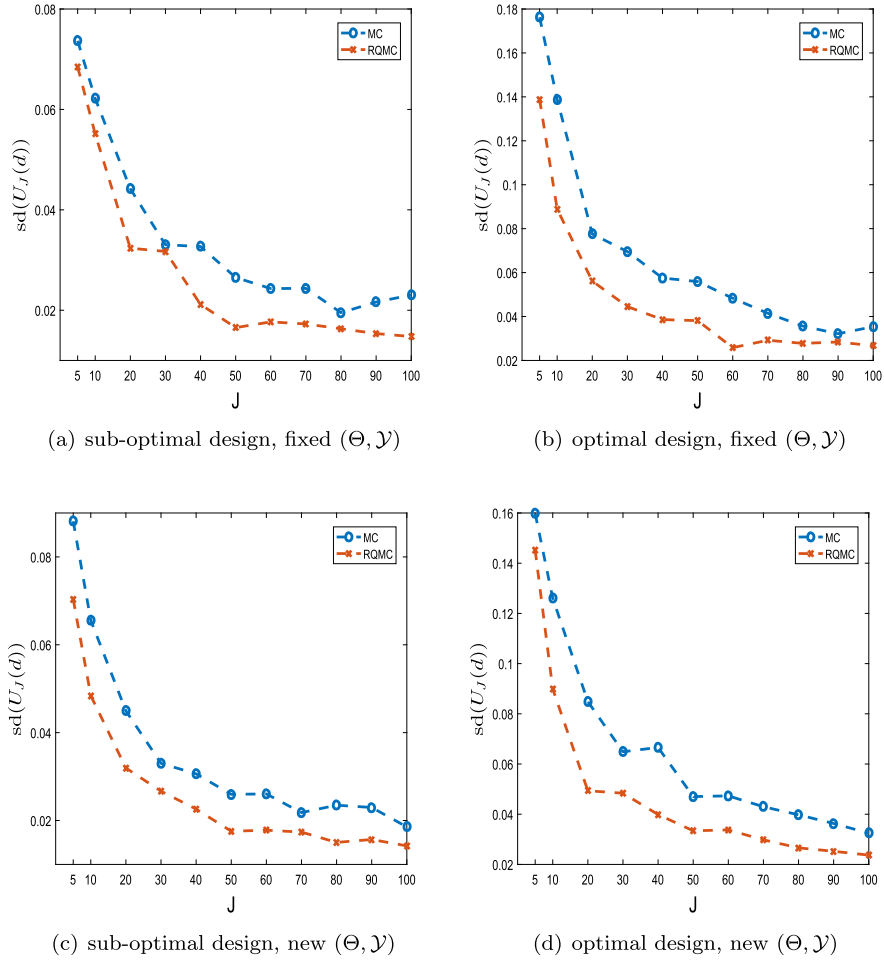
Figure 8: Estimated standard deviations of the estimated expected utilities $U_J(\boldsymbol{d})$ for different values of J for the discrimination example. Subfigures (a) and (c) show results for a sub-optimal design whilst subfigures (b) and (d) show results for a more optimal design. Subfigures (a) and (b) use the same importance samples to estimate $U(\boldsymbol{d}, \boldsymbol{y})$ for different $\boldsymbol{y}$ whilst subfigures (c) and (d) generate new importance samples for each $\boldsymbol{y}$. Results based on MC integration are shown as circles whilst results based on RQMC are shown as crosses. Points are connected via linear interpolation.

that precision can be gained without compromising at all on computational cost. In high dimensional design optimisation problems, we demonstrated that RQMC can more rapidly find designs with high expected utility relative to MC.

Monte Carlo variability of the $U(\boldsymbol{d}, \boldsymbol{y})$ estimation will be present regardless of whether MC or RQMC is used to estimate the expected utility, $U(\boldsymbol{d})$, and this could

dampen the impact of RQMC. We did not consider the possibility of implementing RQMC to reduce also the variability of $U_N(\boldsymbol{d}, \boldsymbol{y})$. Given that $U_N(\boldsymbol{d}, \boldsymbol{y})$ is typically some functional of the posterior it is less clear how to draw an RQMC sample from the posterior when it does not have an explicit form. McGree et al. (2016) consider using a Laplace approximation of the posterior as the importance distribution in IS and draw samples from this multivariate normal approximation using RQMC.

References

Berger, J. (1985). *Statistical Decision Theory and Bayesian Analysis*. New York: Springer-Verlag. MR0804611. doi: https://doi.org/10.1007/978-1-4757-4286-2. 141

Bernardo, J. M. and Smith, A. (2000). *Bayesian Theory*. Chichester: Wiley. MR1274699. doi: https://doi.org/10.1002/9780470316870. 139, 157

Bliemer, M. C. J., Rose, J. M., and Hess, S. (2008). "Approximation of Bayesian efficiency in experimental choice designs." *Journal of Choice Modelling*, 1(1): 98–126. 141

Box, G. E. and Muller, M. E. (1958). "A note on the generation of random normal deviates." *The Annals of Mathematical Statistics*, 29(2): 610–611. 145

Box, G. E. P. and Hill, W. J. (1967). "Discrimination among mechanistic models." *Technometrics*, 9: 57–71. MR0223048. 158

Cook, A. R., Gibson, G. J., and Gilligan, C. A. (2008). "Optimal observation times in experimental epidemic processes." *Biometrics*, 64(3): 860–868. MR2526637. doi: https://doi.org/10.1111/j.1541-0420.2007.00931.x. 145, 154

Dehideniya, M. B., Drovandi, C. C., and McGree, J. M. (2016). "Effcient Bayesian Design for Discriminating Between Models with Intractable Likelihoods in Epidemiology." https://eprints.qut.edu.au/97824/. 158

Dick, J. and Pillichshammer, F. (2010). *Digital Nets and Sequence. Discrepancy Theory and Quasi-Monte Carlo Integration*. Cambridge: Cambridge University Press. MR2683394. doi: https://doi.org/10.1017/CBO9780511761188. 142, 144

Drovandi, C. C., McGree, J. M., and Pettitt, A. N. (2013). "Sequential Monte Carlo for Bayesian sequentially designed experiments for discrete data." *Computational Statistics and Data Analysis*, 57: 320–335. MR2981091. doi: https://doi.org/10.1016/j.csda.2012.05.014. 141

Drovandi, C. C., McGree, J. M., and Pettitt, A. N. (2014). "A sequential Monte Carlo algorithm to incorporate model uncertainty in Bayesian sequential design." *Journal of Computational and Graphical Statistics*, 23: 3–24. MR3173758. doi: https://doi.org/10.1080/10618600.2012.730083. 158

Drovandi, C. C. and Pettitt, A. N. (2013). "Bayesian experimental design for models with intractable likelihoods." *Biometrics*, 69(4): 937–948. MR3146789. doi: https://doi.org/10.1111/biom.12081. 145, 154, 155

Gerber, M. and Chopin, N. (2015). "Sequential quasi Monte Carlo." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(3): 509–579. MR3351446. doi: https://doi.org/10.1111/rssb.12104. 141

Gillespie, D. T. (1977). "Exact stochastic simulation of coupled chemical reactions." *The Journal of Physical Chemistry*, 81(25): 2340–2361. 155

Glasserman, P. (2004). *Monte Carlo Methods in Financial Engineering*. New York: Springer-Verlag. MR1999614. 142

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. MR2664866. doi: https://doi.org/10.1198/TECH.2009.0009. 146

Hainy, M., Müller, W. G., and Wagner, H. (2014). "Likelihood-Free Simulation-Based Optimal Design: An Introduction." In Melas, V., Mignani, S., Monari, P., and Salmaso, L. (eds.), *Topics in Statistical Simulation*, 271–278. Springer. 154

Halton, J. H. (1960). "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals." *Numerische Mathematik*, 2: 84–90. MR0121961. 143

Hammersley, J. M. (1960). "Monte Carlo methods for solving multivariable problems." *Annals of the New York Academy of Sciences*, 86: 844–874. MR0117870. 143

Huan, X. and Marzouk, Y. M. (2014). "Gradient-based stochastic optimization methods in Bayesian experimental design." *International Journal for Uncertainty Quantification*, 4(6): 479–510. MR3275924. doi: https://doi.org/10.1615/Int.J.UncertaintyQuantification.2014006730. 146

Loh, W.-L. (2003). "On the asymptotic distribution of scrambled net quadrature." *Annals of Statistics*, 31: 1282–1324. MR2001651. doi: https://doi.org/10.1214/aos/1059655914. 145

Matousek, J. (1998). "On the L2-discrepancy for anchored boxes." *Journal of Complexity*, 14: 527–556. MR1659004. doi: https://doi.org/10.1006/jcom.1998.0489. 144

McGree, J. M., Drovandi, C. C., White, G., and Pettitt, A. N. (2016). "A pseudo-marginal sequential Monte Carlo algorithm for random effects models in Bayesian sequential design." *Statistics and Computing*, 26(5): 1121–1136. MR3535048. doi: https://doi.org/10.1007/s11222-015-9596-z. 160

Meyer, R. K. and Nachtsheim, C. J. (1995). "The coordinate-exchange algorithm for constructing exact optimal experimental designs." *Technometrics*, 37: 60–69. MR1322047. doi: https://doi.org/10.2307/1269153. 146

Moler, C. and van Loan, C. (2003). "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later." *SIAM Review*, 45(1): 3–49. MR1981253. doi: https://doi.org/10.1137/S00361445024180. 154

Müller, P., Sansó, B., and De Iorio, M. (2004). "Optimal Bayesian design by inhomogeneous Markov chain simulation." *Journal of the American Statisti-

cal Association, 99(467): 788–798. MR2090911. doi: https://doi.org/10.1198/016214504000001123.   145

Niederreiter, H. (1992). *Random Number Generation and Quasi-Monte Carlo Methods*. Philadelphia: Society for Industrial and Applied Mathematics. MR1172997. doi: https://doi.org/10.1137/1.9781611970081.   142, 143, 144

Overstall, A. M., McGree, J. M., and Drovandi, C. C. (2016). "Fully Bayesian optimal design using the approximate coordinate exchange algorithm and normal-based approximations to posterior quantities." https://arxiv.org/abs/1608.05815.   140, 148, 149

Overstall, A. M. and Woods, D. C. (2016). "Bayesian Design of Experiments using Approximate Coordinate Exchange." arXiv:1501.00264.   146, 149

Owen, A. B. (1997). "Scrambled net variance for integrals of smooth functions." *Annals of Statistics*, 25: 1541–1562. MR1463564. doi: https://doi.org/10.1214/aos/1031594731.   141, 144

Robert, C. P. (2007). *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer Science & Business Media. MR2723361. 141

Ryan, E. G., Drovandi, C. C., McGree, J. M., and Pettitt, A. N. (2016). "A review of modern computational algorithms for Bayesian optimal design." *International Statistical Review*, 84: 128–154. MR3491282.   140

Ryan, E. G., Drovandi, C. C., and Pettitt, A. N. (2015). "Fully Bayesian experimental design for Pharmacokinetic studies." *Entropy*, 17: 1063–1089.   148

Ryan, E. G., Drovandi, C. C., Thompson, M. H., and Pettitt, A. N. (2014). "Towards Bayesian experimental design for nonlinear models that require a large number of sampling times." *Computational Statistics and Data Analysis*, 70: 45–60. MR3125477. doi: https://doi.org/10.1016/j.csda.2013.08.017.   147

Tran, M., Nott, D. J., and Kohn, R. (2015). "Variational Bayes with Intractable Likelihood." arXiv:1503.08621.   141

van der Corput, J. G. (1935). "Verteilungsfunktionen. I. Mitt." *Proceedings. Akadamie van Wetenschappen Amsterdam (in German)*, 38: 813–821.   143

Weaver, B. P., Williams, B. J., Anderson-Cook, C. M., and Higdon, D. M. (2016). "Computational enhancements to Bayesian design of experiments using Gaussian processes." *Bayesian Analysis*, 11(1): 191–213. MR3447096. doi: https://doi.org/10.1214/15-BA945.   146

Acknowledgments