Bayesian Analysis (2020) **TBA**, Number TBA, pp. 1–30

Sequential Bayesian Experimental Design for Implicit Models via Mutual Information

Steven Kleinegesse[*], Christopher Drovandi[†], and Michael U. Gutmann[‡]

Abstract. Bayesian experimental design (BED) is a framework that uses statistical models and decision making under uncertainty to optimise the cost and performance of a scientific experiment. Sequential BED, as opposed to static BED, considers the scenario where we can sequentially update our beliefs about the model parameters through data gathered in the experiment. A class of models of particular interest for the natural and medical sciences are implicit models, where the data generating distribution is intractable, but sampling from it is possible. Even though there has been a lot of work on static BED for implicit models in the past few years, the notoriously difficult problem of sequential BED for implicit models has barely been touched upon. We address this gap in the literature by devising a novel sequential design framework for parameter estimation that uses the Mutual Information (MI) between model parameters and simulated data as a utility function to find optimal experimental designs, which has not been done before for implicit models. Our approach uses likelihood-free inference by ratio estimation to simultaneously estimate posterior distributions and the MI. During the sequential BED procedure we utilise Bayesian optimisation to help us optimise the MI utility. We find that our framework is efficient for the various implicit models tested, yielding accurate parameter estimates after only a few iterations.

Keywords: Bayesian experimental design, likelihood-free inference, mutual information, approximate Bayesian computation, implicit models.

MSC 2010 subject classifications: Primary 62K05; secondary 62L05.

1 Introduction

Scientific experiments are critical to improving our perception and understanding of how the world works. Most of the time these experiments are time-consuming and expensive to perform. It is thus crucial to decide where and how to collect the necessary data to learn most about the subject of study. *Bayesian experimental design* attempts to solve this problem by allocating resources in an experiment using Bayesian statistics (see e.g. Ryan et al., 2016 for a comprehensive review). Roughly speaking, the aim is to find experimental designs, e.g. measurement times or locations, that are expected to most rapidly address the scientific aims of the experiment, mitigating the costs. The relevant scientific objectives include model parameter estimation, prediction of future observations or comparison of competing models. For instance, if we wanted to

[*]University of Edinburgh, steven.kleinegesse@ed.ac.uk
[†]Queensland University of Technology, c.drovandi@qut.edu.au
[‡]University of Edinburgh, michael.gutmann@ed.ac.uk

 https://doi.org/10.1214/20-BA1225

estimate the infection rate of a disease, we could use Bayesian experimental design to find out when or where we should count the number of infected individuals in a population.

At the core of Bayesian experimental design is the so-called utility function, which is maximised to find the optimal design at which to perform an experiment. A popular and principled utility function for parameter estimation is the *mutual information* between model parameters and simulated data (Lindley, 1972). Intuitively, this metric measures the additional information we would obtain about the model parameters given some real-world observations taken at a particular design. But depending on the model, computing the mutual information can be difficult or even intractable.

Whenever new, real-world data is collected through physical experiments, the surface of the utility function tends to change, e.g. collecting data with the same design would generally not yield much new information. The treatment of this change, for a single, new data point, is called myopic *sequential* Bayesian experimental design and is manifested through an update of the prior distribution upon observing real-world data. This stands in contrast to *static* Bayesian experimental design that is concerned with situations where we do not update our design strategy when observing new data, such as when there is nearly no time, or too much time, between real-world measurements, or data has to be collected all at once. Sequential Bayesian experimental design is a well-established field for situations in which the model has a tractable likelihood function and inferring the posterior distribution is straight-forward (Ryan et al., 2016). However, there have only been few studies (e.g. Hainy et al., 2016) pertaining to the arguably more realistic situation of intractable, *implicit models*.

In practice, statistical models commonly have likelihood functions that are analytically unknown or intractable. This is the case for implicit models (Diggle and Gratton, 1984), where we cannot evaluate the likelihood but sampling is possible. They are ubiquitous in the natural and medical sciences and therefore have widespread use. Examples include ecology (Ricker, 1954; Wood, 2010), epidemiology (Numminen et al., 2013; Corander et al., 2017), genetics (Marttinen et al., 2015; Arnold et al., 2018), cosmology (Schafer and Freeman, 2012; Alsing et al., 2018) and modelling particle collisions (Agostinelli et al., 2003; Sjöstrand et al., 2008). Implicit models have been used in the context of Bayesian experimental design, for example, to model local weather (Hainy et al., 2015) or the pharmakinetics of drug administration (Overstall and Woods, 2017). We will here consider implicit models from epidemiology (Allen, 2008) and a model of the spread of cells on a scratch assay (Vo et al., 2015).

In order to compute the mutual information between model parameters and simulated data, one needs to be able to evaluate the ratio between posterior density to prior density several times which is difficult in the likelihood-free setting. This is especially challenging in the sequential framework, where the current belief distribution gets updated after every observation. In this work we propose to approximate the density ratio in mutual information directly via the Likelihood-Free Inference by Ratio Estimation (LFIRE) method of Thomas et al. (2016). We perform this in the context of sequential Bayesian experimental design, a significant extension of Kleinegesse and Gutmann (2019) that only considered the static setting.

In this paper we propose a sequential Bayesian experimental design framework for implicit models that have intractable data-generating distributions.[1] In brief, we make the following contributions:

1. Our approach allows us to approximate the mutual information in the presence of an implicit model directly by LFIRE, without resorting to simulation-based likelihood approximations required by other approaches. At the same time, LFIRE also provides an approximation of the sequential posterior.

2. We demonstrate the efficacy of our sequential framework on examples from epidemiology and cell biology. We further showcase that previous approaches may produce experimental designs that heavily penalise multi-modal posteriors thereby introducing an undesirable bias into the scientific data gathering stage, which our approach avoids.

In Section 2 we give basic background knowledge to sequential Bayesian experimental design, mutual information and likelihood-free inference, in particular LFIRE. We then combine these concepts in Section 3 and explain our novel framework of sequential design for implicit models. We test our framework on various implicit models and present the results in Section 4. We conclude our work and discuss possible future work in Section 5.

2 Background

2.1 Bayesian Experimental Design

In Bayesian experimental design (BED) the aim is to find experimental designs $\mathbf{d}$ that yield more informative, or useful, real-world observations than others. Furthermore, in this work we are particularly interested in finding the optimal design $\mathbf{d}^*$ that results in the best estimation of the model parameters. At its core, this task requires defining a utility function $U(\mathbf{d})$ that describes the value of performing an experiment at $\mathbf{d} \in \mathcal{D}$, where $\mathcal{D}$ defines the space of possible designs. In order to qualify as a 'fully Bayesian design', this utility has to be a functional of the posterior distribution $p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y})$ (Ryan et al., 2016), where $\boldsymbol{\theta}$ are the model parameters and $\mathbf{y}$ is simulated data. The utility function is then maximised in order to find the optimal design $\mathbf{d}^*$, i.e.

$$\mathbf{d}^* = \underset{\mathbf{d} \in \mathcal{D}}{\arg\max}\, U(\mathbf{d}). \tag{2.1}$$

The choice of utility function $U(\mathbf{d})$ is thus critical, as different functions will usually lead to different optimal designs. The most suitable utilities naturally depend on the task in question, but there are a few common functions that have been used extensively in the literature. For instance, the *Bayesian D-Optimality* (BD-Opt) is based on the determinant of the inverse covariance matrix of the posterior distribution,[2] and is a

[1] The corresponding code can be found at: https://github.com/stevenkleinegesse/seqbed.
[2] See Appendix A (Kleinegesse et al., 2020) for an alternative form of the BD-Opt utility.

measure of how precise the resulting posterior might be (Ryan et al., 2016),

$$U(\mathbf{d}) = \mathbb{E}_{p(\mathbf{y}|\mathbf{d})}\left[\frac{1}{\det(\mathrm{cov}(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{d}))}\right]. \tag{2.2}$$

While BD-Opt works well for uni-modal posteriors, it is not suitable for multi-modal or complex posteriors as it heavily penalises diversion from uni-modality. A more versatile and robust utility function is the *mutual information*, one of the most principled choices in Bayesian experimental design (e.g. Ryan et al., 2016).

2.2 Mutual Information

The mutual information $\mathrm{I}(\boldsymbol{\theta}; \mathbf{y}|\mathbf{d})$ can be interpreted as the expected reduction in uncertainty (entropy) of the model parameters if the data $\mathbf{y}$ was obtained with design $\mathbf{d}$. It accounts for possibly non-linear dependencies between $\boldsymbol{\theta}$ and $\mathbf{y}$. It is an effective metric with regards to the task of parameter estimation, as we are essentially concerned with finding the design for which the corresponding observation yields the most information about the model parameters $\boldsymbol{\theta}$. In other words, mutual information tells us how 'much' we can learn about the model parameters given the prospective data at a particular design.

Mutual information is defined as the Kullback-Leibler (KL) divergence D_{KL} (Kullback and Leibler, 1951) between the joint distribution and the product of marginal distributions of $\mathbf{y}$ and $\boldsymbol{\theta}$ given $\mathbf{d}$, i.e.

$$\begin{aligned}
\mathrm{I}(\boldsymbol{\theta}; \mathbf{y} \mid \mathbf{d}) &= \mathrm{D}_{\mathrm{KL}}(p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d}) \mid\mid p(\boldsymbol{\theta} \mid \mathbf{d}) p(\mathbf{y} \mid \mathbf{d})) && (2.3)\\
&= \mathrm{D}_{\mathrm{KL}}(p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d}) \mid\mid p(\boldsymbol{\theta}) p(\mathbf{y} \mid \mathbf{d})) && (2.4)\\
&= \int p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d}) \log \left[\frac{p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d})}{p(\boldsymbol{\theta}) p(\mathbf{y} \mid \mathbf{d})}\right] \mathrm{d}\boldsymbol{\theta}\mathrm{d}\mathbf{y}, && (2.5)
\end{aligned}$$

where we have made the usual assumption that our prior belief about $\boldsymbol{\theta}$ is not affected by the design, i.e. $p(\boldsymbol{\theta} \mid \mathbf{d}) = p(\boldsymbol{\theta})$.

The mutual information can also be interpreted as the expected KL divergence between posterior $p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y})$ and prior $p(\boldsymbol{\theta})$ (see e.g. Ryan et al., 2016) and essentially tells us how different on average our posterior distribution is to the prior distribution. The utility that we then need to maximise in order to find the optimal design $\mathbf{d}^*$ is thus

$$\begin{aligned}
U(\mathbf{d}) &= \mathrm{I}(\boldsymbol{\theta}; \mathbf{y} \mid \mathbf{d}) && (2.6)\\
&= \mathbb{E}_{p(\mathbf{y}|\mathbf{d})}[\mathrm{D}_{\mathrm{KL}}(p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}) \mid\mid p(\boldsymbol{\theta}))] && (2.7)\\
&= \int \log \left[\frac{p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y})}{p(\boldsymbol{\theta})}\right] p(\boldsymbol{\theta}) p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) \mathrm{d}\boldsymbol{\theta}\mathrm{d}\mathbf{y}, && (2.8)
\end{aligned}$$

where $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$ is the data generating distribution. The expression for mutual information in (2.8) can be obtained from (2.5) by applying the product rule to $p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d})$. Even though mutual information is a well-studied concept, estimating it efficiently remains an open question, especially in higher dimensions.

Assuming that we have optimised the utility function in (2.8) and have obtained the optimal design $\mathbf{d}^*$, an experimenter would then next go and perform the experiment at $\mathbf{d}^*$ and observe real-world data $\mathbf{y}^*$. Everything up to this point is *static* Bayesian experimental design. If we would like to update our optimal design in light of the real-world observation, we would have to perform *sequential* Bayesian experimental design, i.e. update our prior distribution and optimise the utility function again. This procedure is then repeated several times to obtain (myopic) sequentially designed experiments. We shall in this paper not aim to find non-myopic sequentially designed experiments where we would we plan ahead more than one time-step as this adds another layer of complexity.

Let k be the kth iteration of the sequential design procedure, where $k = 1$ corresponds to the task of finding the first optimal experimental design $\mathbf{d}_1^*$ yielding real-world observation $\mathbf{y}_1^*$. At iteration k we then optimise the utility function $U_k(\mathbf{d})$ to obtain sequential optimal designs $\mathbf{d}_k^*$, with corresponding real-world observations $\mathbf{y}_k^*$. The utility function at iteration $k \in \{1, 2, \dots, K\}$ depends on the set of all previous observations $\mathbb{D}_{k-1} = \{\mathbf{d}_{1:k-1}^*, \mathbf{y}_{1:k-1}^*\}$, with $\mathbb{D}_0 = \varnothing$, and therefore will change at every iteration. Its form stays similar to (2.8), except that the prior and posterior distributions now depend on $\mathbb{D}_{k-1}$, i.e.

$$U_k(\mathbf{d}) = \int \log \left(\frac{p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}, \mathbb{D}_{k-1})}{p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})} \right) p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{y}. \tag{2.9}$$

Note that we will assume that data is generated independently of previous observations, i.e. $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}, \mathbb{D}_{k-1}) = p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$. This would not hold in special cases when gathering real-world observations changes the data-generating process.

2.3 Likelihood-Free Inference

Implicit models have intractable likelihood functions, which means that $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$ is either too expensive to compute or there is no closed-form expression. This results in standard Bayesian inference becoming infeasible. Because of their widespread use however, it is crucial to be able to infer the parameters of implicit models. As a result, the field of *likelihood-free inference* has emerged. These methods leverage the fact that, by definition, implicit models allow for sampling from the data-generating distribution.

A popular likelihood-free approach is Approximate Bayesian Computation (ABC, Rubin, 1984). ABC rejection sampling (Pritchard et al., 1999), the simplest form of ABC, works by generating samples from the prior distribution over the model parameters and then using them to simulate data from the implicit model. The prior parameters that result in data that is 'close' to observed data are then accepted as samples from the ABC posterior distribution. See Sisson et al. (2018) or Lintusaari et al. (2017) for reviews on ABC.

Since standard ABC is notoriously slow and requires tuning of some hyperparameters, there has been considerable research in making likelihood-free inference more efficient, using, for example, ideas from Bayesian optimisation and experimental design (Gutmann and Corander, 2016; Järvenpää et al., 2019, 2020), conditional density estimation (Papamakarios and Murray, 2016; Lueckmann et al., 2017; Greenberg

et al., 2019), classification (Gutmann et al., 2018), indirect inference (Drovandi et al., 2015), optimisation (Meeds and Welling, 2015; Ikonomov and Gutmann, 2020), and more broadly surrogate modelling with Gaussian processes (Wilkinson, 2014; Meeds and Welling, 2015) and neural networks (Blum and Francois, 2010; Chen and Gutmann, 2019; Papamakarios et al., 2019).

In this paper, we make use of another approach to likelihood-free inference called Likelihood-Free Inference by Ratio Estimation (LFIRE, Thomas et al., 2016). LFIRE uses density ratio estimation to obtain ratios $r(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta})$ of the likelihood to marginal density and, therefore, the posterior to prior density, i.e.

$$r(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}) = \frac{p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})}{p(\mathbf{y} \mid \mathbf{d})} = \frac{p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y})}{p(\boldsymbol{\theta})}. \tag{2.10}$$

The method works by estimating the ratio from data simulated from $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$ and $p(\mathbf{y} \mid \mathbf{d})$, e.g. via logistic regression (Thomas et al., 2016). Since the prior density $p(\boldsymbol{\theta})$ is known, learning the ratio corresponds to learning the posterior, i.e. $\widehat{p}(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}) = \widehat{r}(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}) p(\boldsymbol{\theta})$. Importantly, the learned ratio yields automatically also an estimate of the mutual information in (2.9).

The LFIRE framework can be used with arbitrary models of the ratio or posterior. For simplicity, like in the simulations by Thomas et al. (2016), we here use the log-linear model

$$\widehat{r}(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}) = \exp\left(\boldsymbol{\beta}(\mathbf{d}, \boldsymbol{\theta})^\top \boldsymbol{\psi}(\mathbf{y})\right), \tag{2.11}$$

where $\boldsymbol{\psi}(\mathbf{y})$ are some fixed summary statistics. Thomas et al. (2016) showed that this log-linear model, while simple, generalises the popular synthetic likelihood approach by Wood (2010); Price et al. (2018b). Moreover, learning the summary statistics from data, e.g. by means of neural networks, is possible too (Dinev and Gutmann, 2018). For further details on LFIRE, we refer the reader to the original paper by Thomas et al. (2016).

3 Sequential Mutual Information Estimation

The main aim of this work is to construct an effective sequential experimental design framework for implicit models. To do this, we have to approximate the sequential utility in (2.9) in a tractable manner. We propose to use LFIRE to estimate the intractable density ratio in (2.9) and, at the same time, obtain the posterior density. The main difference to the work of Kleinegesse and Gutmann (2019) is that they only considered static experimental design and did not have the additional complications that come with the sequential setting, such as updating the prior distribution upon observing real-world data. Our approach bears some similarities to the Sequential Monte-Carlo (SMC) design method of Hainy et al. (2016). However, we use LFIRE for updating the posterior when new data are collected and for direct estimation of the mutual information (MI), rather than relying on simulation-based likelihood estimation. Further, unlike Hainy et al. (2016), our approach avoids MCMC for the resampling step. While our approach does not exactly preserve the distribution of the particles, it does not require re-processing all data seen so far, significantly accelerating computation.

3.1 Sequential Utility

We assume that we have already made $k-1$ experiments resulting in the set of optimal designs and observations $\mathbb{D}_{k-1} = \{\mathbf{d}^*_{1:k-1}, \mathbf{y}^*_{1:k-1}\}$, with $\mathbb{D}_0 = \varnothing$. At iteration k of the sequential BED procedure we then set out to determine the optimal design $\mathbf{d}^*_k$ and the corresponding real-world observation $\mathbf{y}^*_k$. To do so, we first approximate the density ratio of $p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}, \mathbb{D}_{k-1})$ and $p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})$ by the ratio $\widehat{r}_k(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}, \mathbb{D}_{k-1})$ computed by LFIRE,[3] such that

$$\widehat{r}_k(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}, \mathbb{D}_{k-1}) \approx \frac{p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}, \mathbb{D}_{k-1})}{p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})}. \tag{3.1}$$

We then plug this into the expression for the sequential MI utility in (2.9) and obtain

$$U_k(\mathbf{d}) = \int \log\left(\frac{p(\boldsymbol{\theta} \mid \mathbf{d}, \mathbf{y}, \mathbb{D}_{k-1})}{p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})}\right) p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1}) \mathrm{d}\boldsymbol{\theta}\mathrm{d}\mathbf{y} \tag{3.2}$$

$$\approx \int \log\left(\widehat{r}_k(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}, \mathbb{D}_{k-1})\right) p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1}) \mathrm{d}\boldsymbol{\theta}\mathrm{d}\mathbf{y}. \tag{3.3}$$

We can approximate this with a Monte-Carlo sample average to obtain the estimate

$$\widehat{U}_k(\mathbf{d}) = \frac{1}{N} \sum_{i=1}^{N} \log\left[\widehat{r}_k(\mathbf{d}, \mathbf{y}^{(i)}, \boldsymbol{\theta}^{(i)}, \mathbb{D}_{k-1})\right], \tag{3.4}$$

where $\mathbf{y}^{(i)} \sim p(\mathbf{y} \mid \mathbf{d}, \boldsymbol{\theta}^{(i)})$ and $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})$. The above mutual information estimate $\widehat{U}_k(\mathbf{d})$ is then optimised to find the optimal design $\mathbf{d}^*_k$ and, through a real-world experiment, the corresponding observation $\mathbf{y}^*_k$ at iteration k.

Two core technical difficulties in (3.4) are (1) how to obtain parameter samples $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})$ from the updated belief distribution and (2) how to compute the sequential LFIRE ratio in (3.1) given the observations $\mathbb{D}_{k-1}$. We explain our solutions to these difficulties in Sections 3.2 to 3.4.

3.2 Updating the Belief About the Model Parameters

For iteration $k = 1$ we only require samples from the prior distribution $p(\boldsymbol{\theta})$ in order to compute the MI in (3.4). We here assume that sampling from the prior is possible. For iteration $k = 2$, we require samples from $p(\boldsymbol{\theta} \mid \mathbb{D}_1)$, for $k = 3$ we require samples from $p(\boldsymbol{\theta} \mid \mathbb{D}_2)$, etc. We here describe how to obtain samples from the updated belief $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$ after any iteration k. For that, let us first define what it means to update the belief about the model parameters. After observing real-word data $\mathbf{y}^*_k$ at optimal design $\mathbf{d}^*_k$, we update the observation data set, i.e. $\mathbb{D}_k = \mathbb{D}_{k-1} \cup \{\mathbf{d}^*_k, \mathbf{y}^*_k\}$. For $\mathbf{d} = \mathbf{d}^*_k$ and $\mathbf{y} = \mathbf{y}^*_k$, the numerator in (3.1) equals $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$, leading us to an expression for the updated belief distribution,

$$p(\boldsymbol{\theta} \mid \mathbb{D}_k) \approx \widehat{r}_k(\mathbf{d}^*_k, \mathbf{y}^*_k, \boldsymbol{\theta}, \mathbb{D}_{k-1}) p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1}). \tag{3.5}$$

[3] Note that LFIRE actually estimates the log ratio of posterior to prior density.

Furthermore, we can approximate the belief distribution $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$ after iteration k as a product of k estimated density ratios and the initial prior $p(\boldsymbol{\theta})$,

$$p(\boldsymbol{\theta} \mid \mathbb{D}_k) \approx \widehat{r}_k(\mathbf{d}_k^*, \mathbf{y}_k^*, \boldsymbol{\theta}, \mathbb{D}_{k-1}) \cdots \widehat{r}_1(\mathbf{d}_1^*, \mathbf{y}_1^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}). \tag{3.6}$$

Each of the density ratios $\widehat{r}_s$ in (3.6) are evaluated at the observations $\{\mathbf{d}_s^*, \mathbf{y}_s^*\}$ of the relevant iteration s, but also depend on all previous observations $\mathbb{D}_{s-1}$. We can write this product of density ratios as a weight function w_k and then (3.6) becomes

$$p(\boldsymbol{\theta} \mid \mathbb{D}_k) \approx w_k(\boldsymbol{\theta}; \mathbb{D}_k) p(\boldsymbol{\theta}), \tag{3.7}$$

where we have defined the weight function w_k to be

$$w_k(\boldsymbol{\theta}; \mathbb{D}_k) = \prod_{s=1}^{k} \widehat{r}_s(\mathbf{d}_s^*, \mathbf{y}_s^*, \boldsymbol{\theta}, \mathbb{D}_{s-1}), \tag{3.8}$$

with $\widehat{r}_1(\mathbf{d}_1^*, \mathbf{y}_1^*, \boldsymbol{\theta}, \mathbb{D}_0) = \widehat{r}_1(\mathbf{d}_1^*, \mathbf{y}_1^*, \boldsymbol{\theta})$ according to (2.10) and $w_0(\boldsymbol{\theta}) = 1 \, \forall \, \boldsymbol{\theta}$.

We use the weight function in (3.8) to obtain samples from the updated belief distribution $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$. To do so, we first sample N initial prior samples $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$. After every iteration k we then obtain weights $w_k^{(i)} = w_k(\boldsymbol{\theta}^{(i)}; \mathbb{D}_k)$ corresponding to the initial prior samples, which form a particle set $\{w_k^{(i)}, \boldsymbol{\theta}^{(i)}\}_{i=1}^{i=N}$. We compute these by updating each weight $w_{k-1}^{(i)}$ by the LFIRE ratio evaluated at the observed data according to (3.8), in order to yield $w_k^{(i)}$. Since we can store the weights $w_{k-1}^{(i)}$ for a particular parameter, we do not need to recompute them.

To finally obtain updated belief samples, we first normalise the weights: $W_k^{(i)} = w_k^{(i)} / \Sigma_{i=1}^N w_k^{(i)}$. Then we sample an index from the categorical distribution, i.e. $I \sim \text{cat}(\{W_k^{(i)}\})$, and choose the initial prior sample $\boldsymbol{\theta}^{(I)}$. Repeating this several times results in a set of parameter samples that follows $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$. We have summarised this procedure in Algorithm 1.

Algorithm 1 Obtaining samples from the updated belief $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$.

1: After iteration k, obtain particle set $\{w_k^{(i)}, \boldsymbol{\theta}^{(i)}\}_{i=1}^{i=N}$
2: Normalise the weights: $W_k^{(i)} = w_k^{(i)} / \Sigma_{i=1}^N w_k^{(i)}$ for $i = 1, \ldots, N$
3: **for** $i = 1$ to $i = N$ **do**
4: Sample from a categorical distribution: $I \sim \text{cat}(\{W_k^{(i)}\})$
5: Choose $\boldsymbol{\theta}^{(I)}$ as a sample from the updated prior distribution
6: **end for**

We note that approximating (3.3) directly with weighted samples from $p(\boldsymbol{\theta})$ instead of using Algorithm 1 to obtain samples from $p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})$ would theoretically result in a lower variance Monte-Carlo estimator. However, because we did not observe this in our simulations and Algorithm 1 had lower computation times, we opted to use Algorithm 1 instead. For more details see Appendix B.

3.3 Resampling

We can see from (3.8) that a weight $w_k^{(i)}$ is computed by the product of LFIRE ratios given all previous observations. Less significant parameter samples, i.e. those with a low density ratio, thus have weights that quickly decay to zero. This means that after several iterations we may be left with only a few weights $w_k^{(i)}$ that are effectively non-zero. Eventually, only few different parameter samples $\boldsymbol{\theta}^{(i)}$ of the current particle set are chosen in the sampling scheme in Algorithm 1, which increases the Monte-Carlo error of the sequential utility in (3.4) and of the marginal samples in the sequential LFIRE procedure (see Section 3.4).

We can quantify how many effective samples we have via the *effective sample size* η (Kish, 1965),

$$\eta = \frac{\left(\sum_{i=1}^N w_k^{(i)}\right)^2}{\sum_{i=1}^N \left(w_k^{(i)}\right)^2}. \tag{3.9}$$

If η is small, i.e. $\eta \ll N$, then the samples do not cover much relevant parameter space and our Monte-Carlo approximations may become poor. Thus, if the effective sample size becomes smaller than a minimum sample size $\eta_{\min}$ we need to resample our set of parameter samples; this allows us to have a set of new parameter samples that well-represent the current belief distribution. Throughout this work we shall use the typical value of $\eta_{\min} = N/2$ (e.g. Chen, 2003; Doucet and Johansen, 2009).

For the resampling, we operate in a linearly re-scaled parameter space such that all re-scaled parameter samples $\boldsymbol{\theta}'^{(i)}$ have values between 0 and 1. This simplifies the re-sampling procedure because we will only have to determine one free parameter value, namely σ (see below). If the parameter space has boundary conditions $\mathcal{B}$, through e.g. a bounded prior distribution on $\boldsymbol{\theta}$, then we transform these in the same way as the parameter space, $\mathcal{B} \to \mathcal{B}'$.[4] In this re-scaled $\boldsymbol{\theta}'$ space, we model the belief distribution $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$ after the current iteration k as a truncated Mixture of Gaussians (MoG), i.e.

$$p(\boldsymbol{\theta}' \mid \mathbb{D}_k) \propto \sum_{i=1}^N W_k^{(i)} \mathcal{N}(\boldsymbol{\theta}'; \boldsymbol{\theta}'^{(i)}, \mathbb{I}\sigma^2) \mathbb{1}_{\mathcal{B}'}(\boldsymbol{\theta}'), \tag{3.10}$$

where $\mathbb{I}$ is the identity matrix, σ^2 is a free variance parameter that we need to determine and $W_k^{(i)}$ are the (normalised) weights summing to one. The indicator function $\mathbb{1}_{\mathcal{B}'}(\boldsymbol{\theta}')$ is 1 if $\boldsymbol{\theta}'$ satisfies the boundary conditions $\mathcal{B}'$ and is 0 otherwise. Because of the possible truncation, the right-hand side in the above formula may not be a properly normalised density. Since we will only sample from $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$, we do not need to know the value of the normalising constant. By the law of transformation of random variables, the density in the original parameter space $\boldsymbol{\theta}$ is also a truncated mixture of Gaussians but with a covariance matrix that takes the possibly different scales of the elements of $\boldsymbol{\theta}$ into account (see Appendix C).

[4] See Appendix C for details on the re-scaling transformation.

In (3.10), we have one Gaussian for every parameter sample of the current particle set; each Gaussian is centred at that parameter sample $\boldsymbol{\theta}'^{(i)}$ and has the same standard deviation σ. The parameter σ is typically small which means that (3.10) can be viewed as a type of kernel density estimate but other density estimators that allow for sampling, even fully-parametric ones, could be used as well.

To determine σ, we use a k-dimensional (KD) tree (Bentley, 1975) to first find the nearest neighbour $\text{NN}(\boldsymbol{\theta}'^{(i)})$ of each parameter sample. Let δ be the median of all the distances of a sample to its nearest neighbour, i.e. $\delta = \text{median}\left(|\boldsymbol{\theta}'^{(i)} - \text{NN}(\boldsymbol{\theta}'^{(i)})|\right)$. We then compute the standard deviation σ as a function g of δ, i.e.

$$\sigma = g(\delta), \tag{3.11}$$

where we choose g to be the square-root function in order to increase robustness to possibly large median distances.[5]

In order to get a new sample from the updated belief distribution, we sample from $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$ in (3.10) and then map the sample back to the original parameter space. In more detail, we first sample an index from a categorical distribution, i.e. $I \sim \text{cat}(\{W_k^{(i)}\})$, and then draw a parameter sample from the corresponding Gaussian $\mathcal{N}(\boldsymbol{\theta}'; \boldsymbol{\theta}'^{(I)}, \mathbb{I}\sigma^2)$. We accept this parameter sample if it satisfies the boundary conditions $\mathcal{B}'$ and reject it otherwise. Doing this a number of times yields a set of new parameter samples of equal weight that we then transform back to the original parameter space, i.e. $\boldsymbol{\theta}' \to \boldsymbol{\theta}$, to obtain the desired resampled parameters. Since the transformation does not affect the weights, we set the weight of the resampled parameter samples to be proportional to one. The resampling procedure is summarised in Algorithm 2.

3.4 Sequential LFIRE

As can be seen from (3.1), the sequential LFIRE ratios depend on previous observations. This particular dependency requires us to revise the original LFIRE method of Thomas et al. (2016) slightly. To compute the ratio $\widehat{r}_k(\mathbf{d}, \mathbf{y}, \boldsymbol{\theta}, \mathbb{D}_{k-1})$ we need to sample data from the data generating distribution $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$ and from the marginal $p(\mathbf{y} \mid \mathbf{d}, \mathbb{D}_{k-1})$. We assume that observing data does not affect the data generating process. The marginal distribution, however, does change upon observing data, i.e. at iteration k we have

$$p(\mathbf{y} \mid \mathbf{d}, \mathbb{D}_{k-1}) = \int p(\mathbf{y}, \boldsymbol{\theta} \mid \mathbf{d}, \mathbb{D}_{k-1}) \mathrm{d}\boldsymbol{\theta} \tag{3.12}$$

$$= \int p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1}) \mathrm{d}\boldsymbol{\theta}. \tag{3.13}$$

This implies that in order to obtain samples from the marginal we first have to sample from the belief distribution $p(\boldsymbol{\theta} \mid \mathbb{D}_{k-1})$ according to Algorithm 1. These parameter samples from the updated belief distribution are then plugged into the data generating distribution to finally obtain samples from the marginal. The rest of the LFIRE procedure remains unchanged (see Thomas et al., 2016 for more details).

[5] The log function would work similarly well for this reason.

Algorithm 2 Resampling via a Mixture of Gaussian model.

After iteration k, obtain particle set $\{w_k^{(i)}, \boldsymbol{\theta}^{(i)}\}_{i=1}^{i=N}$
Transform the parameters to be in the unit hyper-cube, $\boldsymbol{\theta} \rightarrow \boldsymbol{\theta}'$
Transform the boundary conditions in the same way, $\mathcal{B} \rightarrow \mathcal{B}'$
Find the nearest neighbour of each parameter sample
Compute the standard deviation σ for the MoG model, according to (3.11)
Normalise the weights: $W_k^{(i)} = w_k^{(i)} / \Sigma_{j=1}^{N} w_k^{(j)}$ for $i = 1, \dots, N$
for $i = 1$ to $i = N$ **do**
    Sample from a categorical distribution: $I \sim \text{cat}(\{W_k^{(i)}\})$
    while not accepted **do**
        Sample $\boldsymbol{\theta}'^{(i)}_{new} \sim \mathcal{N}(\boldsymbol{\theta}'; \boldsymbol{\theta}'^{(I)}, \mathbb{I}\sigma^2)$
        if $\boldsymbol{\theta}'^{(i)}_{new}$ satisfies $\mathcal{B}'$ **then**
            Accept
        else
            Reject
        end if
    end while
end for
Reset the weights to $w_k^{(i)} = 1 \, \forall \, i$
Transform the parameters back to the original parameter space, $\boldsymbol{\theta}'_{new} \rightarrow \boldsymbol{\theta}_{new}$
Return $\{w_k^{(i)}, \boldsymbol{\theta}_{new}^{(i)}\}_{i=1}^{i=N}$

3.5 Optimisation

In all sections hitherto we have explained how to compute the sequential mutual information utility $\widehat{U}_k(\mathbf{d})$ at iteration k. We have, however, not addressed the issue of optimising the utility with respect to the designs $\mathbf{d}$ in order to find the optimal design $\mathbf{d}_k^*$. While traditionally the utility has been optimised via grid search or a sampling-based approach by Müller (1999), there have been a few recent approaches using evolutionary algorithms (Price et al., 2018a) or Gaussian Processes (GP) (Overstall and Woods, 2017). The latter approaches were generally found to outperform grid search in terms of efficiency. We here choose to optimise the sequential utility using Bayesian Optimisation (BO) (Shahriari et al., 2016), as was done by Kleinegesse and Gutmann (2019), due to its flexibility and efficiency. In addition, BO smoothes out the Monte-Carlo error of our utility approximations, and may thus help in locating the optimal design $\mathbf{d}_k^*$ as well.

BO is a popular optimisation scheme for functions that are expensive to evaluate and that potentially have unknown gradients. The general idea is to use a probabilistic surrogate model of the utility and then use a cheaper acquisition function to decide where to evaluate the utility next. We use a GP for the surrogate model with a Matérn-5/2 Kernel (Shahriari et al., 2016) and Expected Improvement (Mockus et al., 1978) for the acquisition function. These are standard choices in the BO literature, for more detail we refer the reader to Shahriari et al. (2016).

We summarise the previous sections by describing our framework of estimating and optimising the sequential mutual information utility in Algorithm 3.

Algorithm 3 Sequential Bayesian Exp. Design via LFIRE using BO.

Let $\mathbb{D}_0 = \varnothing$
Sample initial parameters from prior: $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$ for $i = 1, \ldots, N$
Initialise weights: $w_0^{(i)} = 1$ for $i = 1, \ldots, N$
for $k = 1$ to $k = K$ **do**
    Calculate the effective sampling size η using (3.9)
    if $k = 1$ **then**
        Use all initial prior samples $\{\boldsymbol{\theta}^{(i)}\}_{i=1}^N$
    else if $\eta < \eta_{\min}$ **then**
        Obtain new samples $\{\boldsymbol{\theta}_{new}^{(i)}\}_{i=1}^N$ by resampling according to Algorithm 2
        Set the weights for the iteration to one, i.e. $w_k^{(i)} = 1$ for $i = 1, \ldots, N$
        Use all new parameter samples $\{\boldsymbol{\theta}^{(i)}\}_{i=1}^N \leftarrow \{\boldsymbol{\theta}_{new}^{(i)}\}_{i=1}^N$
    else
        Obtain updated belief samples $\{\boldsymbol{\theta}^{(i)}\}_{i=1}^N$ by applying Algorithm 1
        Use all updated belief samples $\{\boldsymbol{\theta}^{(i)}\}_{i=1}^N$
    end if
    Use BO to determine the maximiser $\mathbf{d}_k^*$ of the sequential utility $\widehat{U}_k(\mathbf{d})$ in (3.4)
    Perform an experiment at $\mathbf{d}_k^*$ to observe some real data $\mathbf{y}_k^*$
    Update the belief distribution by updating the data set: $\mathbb{D}_k = \mathbb{D}_{k-1} \cup \{\mathbf{d}_k^*, \mathbf{y}_k^*\}$
    For all parameter samples $\boldsymbol{\theta}^{(i)}$, compute new weights $w_k^{(i)}$ according to (3.8)
end for

4 Experiments

In this section we test the framework outlined in Algorithm 3 on a number of implicit models from the literature. We first consider an oscillatory toy model with a multi-modal posterior distribution. We then consider the Death Model (Cook et al., 2008) and the Susceptible-Infectious-Recovered (SIR) Model (Allen, 2008) from epidemiology, as well as a model of the spread of cells (Vo et al., 2015).

4.1 Oscillation Toy Model

This toy model describes noisy measurements of a sinusoidal, stationary waveform $\sin(\omega t)$, where the design variable is the measurement time t and the experimental aim is to optimally estimate the waveform's frequency ω. The generative model is given by

$$p(y \mid \omega, t) = \mathcal{N}(y; \sin(\omega t), \sigma_{\text{noise}}^2), \tag{4.1}$$

where we set the measurement noise to $\sigma_{\text{noise}} = 0.1$ throughout and assume that the true model parameter takes a value of $\omega_{\text{true}} = 0.5$. As a prior we use a uniform distribution

$p(\omega) = \mathcal{U}(\omega; 0, \pi)$. We can obtain analytic posterior densities by using the likelihood in (4.1) and Bayes' rule and can obtain corresponding posterior samples by using Markov chain Monte-Carlo (MCMC) methods.

We start the sequential BED procedure for the oscillation model by sampling 1,000 parameter samples $\omega^{(i)}$ from the prior and for each of these we then simulate data $y^{(i)} \sim \mathcal{N}(y; \sin(\omega^{(i)}t), \sigma^2_{\text{noise}})$ at a particular measurement time $t \in [0, 2\pi]$. For the summary statistics in (2.11) we use subsequent powers of the simulated data, i.e. $\boldsymbol{\psi}(y^{(i)}) = [y^{(i)}, (y^{(i)})^2, (y^{(i)})^3]^\top$, in order to allow for a sufficiently flexible, non-linear decision boundary in the LFIRE algorithm. We use these prior samples and the corresponding simulated data to compute 1,000 LFIRE ratios and then estimate the MI utility $\widehat{U}_1(t)$ with a sample average as in (3.4). With the help of BO, we decide at which measurement time t to evaluate the utility next and then repeat until we have maximised the utility, following Algorithm 3.

We repeat the optimisation procedure above for the BD-Opt utility in (2.2) and compare it to the MI utility. Hainy et al. (2016) used this utility in sequential design targeted at implicit models, although they only tested their method on a toy model with known likelihood. The advantages of MI over BD-Opt for models with multi-modal posteriors are widely known in the explicit setting (Ryan et al., 2016). It is nonetheless useful to verify that these advantages continue to hold when approximating the MI and the posterior with LFIRE.

We show the MI utility and the BD-Opt utility used by Hainy et al. (2016), as well as their analytic counterparts, for the first iteration in Figure 1. Shown are the posterior predictive means of the GP surrogate models, the corresponding variances and the evaluations of the utilities during the BO procedure. Due to the chosen prior and the periodic nature of the oscillation model, higher design times result in posterior distributions with more modes. Multi-modality can lead to an increase of the variance. BD-Opt thus assigns little to no worth in doing experiments at late measurement times. In contrast, the MI utility has a high value at late design times when the posterior distributions tend to have multiple modes. The corresponding optimal designs are $t_1^* = 2.196$ and $t_1^* = 1.656$ for the MI utility and the BD-Opt utility, respectively. Furthermore, the behaviour of both utilities generally well matches the analytic references computed using the closed-form expression of the data-generating distribution.

After determining the optimal measurement time t_1^*, we perform the actual experiment. Here, the real-world experiment is simulated by taking a measurement of the true data generating process with $\omega_{\text{true}} = 0.5$ at t_1^* where we obtained $y_1^* = 0.790$ and $y_1^* = 0.810$ for the case of the MI and BD-Opt utility, respectively. We show the corresponding estimates of the posterior distributions in Figure 2. In our approach, we compute particle weights for each of the 1,000 prior samples and then obtain the posterior, or updated belief, samples according to Algorithm 1. Importantly, the BD-Opt utility uses a particle approach as well, which means that we also need to use Algorithm 1 to obtain updated belief samples; we direct the reader to Hainy et al. (2016) for more information on how the required particle weights are computed. For visualisation purposes, we then compute a Gaussian Kernel Density Estimate (KDE) from these posterior samples to obtain the posterior densities shown in Figure 2. We also show the

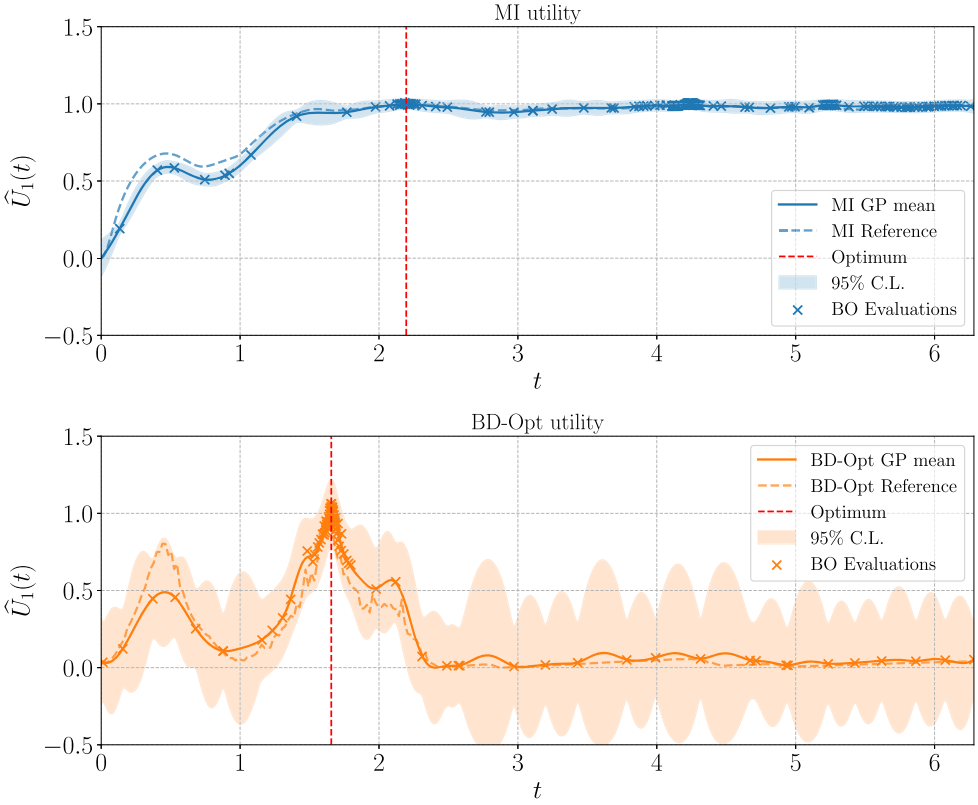

Figure 1: Comparison of MI (top) and BD-Opt (bottom) utilities for the first iteration of the oscillatory toy model, including analytic references.

analytic posteriors which are computed using the closed-form expression of the data-generating distribution. We find that the posterior distributions have two modes, which is a result of the periodic behaviour of the model. We note that one mode has support for the true model parameter.

After obtaining the data $\mathbb{D}_1 = \{\mathbf{d}_1^*, \mathbf{y}_1^*\} = \{t_1^*, y_1^*\}$, we compute the new particle weights $w_1^{(i)}$ via (3.8) for MI and via ABC likelihoods for BD-Opt (see Hainy et al., 2016), which are then used in subsequent iterations of the sequential BED procedure. Following Algorithm 3 we continue this procedure in a similar manner until iteration $k = 4$, although technically this could be continued until the experiment's budget is exhausted.

We show the GP models of the MI and BD-Opt utility for all four iterations in Figure 3. Both utilities change vastly between iterations. As compared to iteration 1, the MI utility has more local optima in iteration 2, although it is still overall increasing and then peaking at $t \approx 6$. A pronounced local minimum occurs around $t = 2.196$, the optimal design of the first iteration; this is intuitive, because performing an experiment

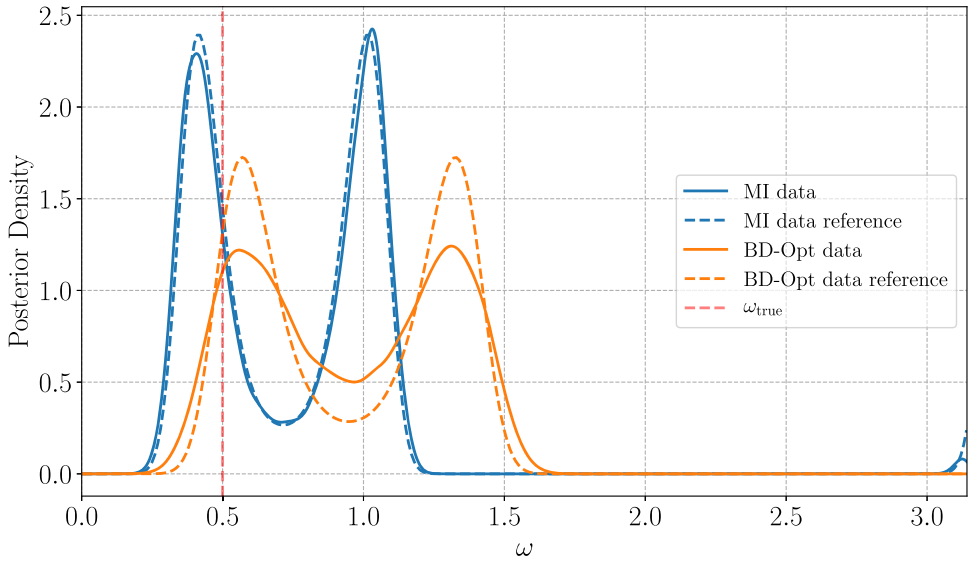

Figure 2: Comparison of the posterior distributions obtained when using the data from the MI (blue) and BD-Opt (orange) utilities for the first iteration of the oscillatory toy model (solid curves), including analytic references (dashed curves).

at the same experimental design may not yield much additional information at this stage due to the relatively small measurement noise. For the same reason, the BD-Opt utility has a local minimum around $t = 1.656$, the optimal design of the first iteration for BD-Opt. Due to large fluctuations in the estimated BD-Opt utility around the global maximum, the GP mean does not go through all nearby evaluations and has a larger variance throughout for iteration 2.

In iteration 3, the MI utility has two local minima that occur at the locations of the two previous optimal designs because, like previously, performing an experiment at the same measurement locations may not be effective. BD-Opt on the other hand steadily increases and then peaks at the upper boundary of the design domain. This occurs because, for BD-Opt, the updated belief distribution of the parameter is uni-modal after iteration 2 and becomes more narrow with increasing design times; similar reasoning follows for BD-Opt in iteration 4. We observe the same behaviour for MI in iteration 4, as the updated belief distribution used to compute the MI utility becomes uni-modal after iteration 3. We had to perform resampling during iterations 2–4, as the effective sample size of the weights went below 50% for both the MI and the BD-Opt utility.[6] Figure 3 shows that in iteration 1 to 3, mutual information assigns worth to several areas in the design domain that Bayesian D-Optimality does not deem important. After enough data is collected and the posterior is unimodal, however, the difference between these two utilities becomes negligible and they result in the same optimal design.

[6]Note that for BD-Opt we use the resampling procedure provided in Hainy et al. (2016).

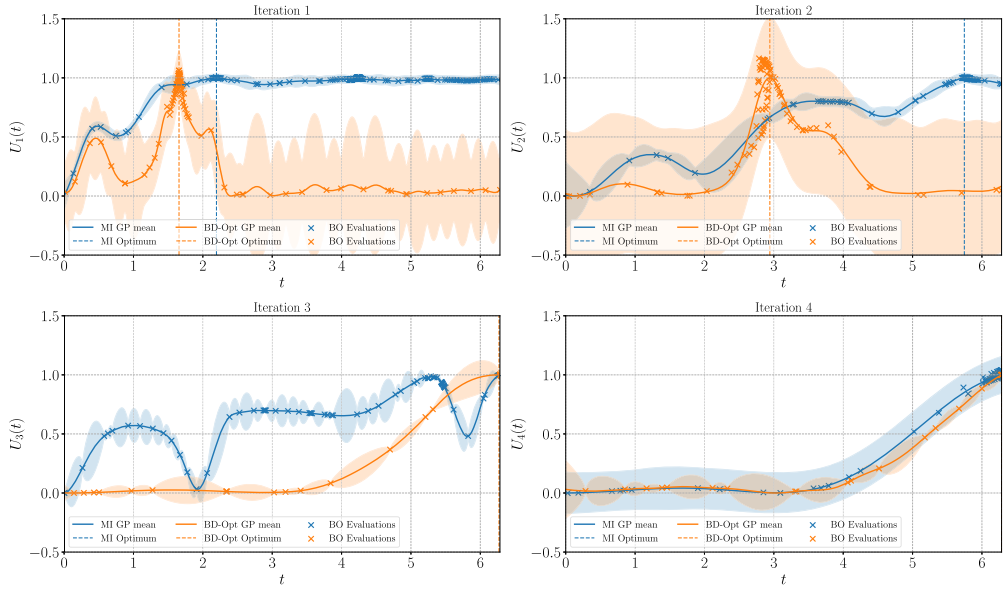

Figure 3: Comparison of the MI and BD-Opt utilities at different iterations for the oscillatory toy model. Shown are the GP means and variances, the BO evaluations and the optima of the GP means. After sufficient iterations, 4 in this case, the difference between the two utilities becomes negligible.

We use KDE to visualise the updated belief samples after each iteration, obtained by means of Algorithm 1. This is shown in Figure 4 for the sequential MI and BD-Opt utilities. After iteration 2, only mutual information results in a multi-modal belief distribution. From iteration 3 onwards, both distributions are unimodal and similarly concentrated around the true model parameter of $\omega_{\text{true}} = 0.5$. After 4 iterations, the mean parameter estimate using the data from the MI utility is $\widehat{\omega} = 0.503$ with a 95% credibility interval of $[0.481, 0.527]$. Using the data from the BD-Opt utility the mean parameter is $\widehat{\omega} = 0.494$ with a 95% credibility interval of $[0.468, 0.516]$. The 95% credibility intervals were computed using a Gaussian KDE of the parameter samples and the highest posterior density interval (HPDI) method.

Overall, in the context of the oscillation model, the mutual information and Bayesian D-Optimality utilities yield significantly different optimal experimental designs. As opposed to MI, BD-Opt leads to optimal experimental designs that are biased to exclude multiple explanations for the inferred parameters. When enough real-world observations are made, the updated belief distributions are no longer multi-modal but collapse to unimodal distributions, at which point the utilities become similar. Additionally, for the BD-Opt utility we noticed certain numerical instabilities that resulted from taking the mean of several posterior precisions. We rectified this by taking the median of several posterior precisions, instead of taking the mean as Hainy et al. (2016).

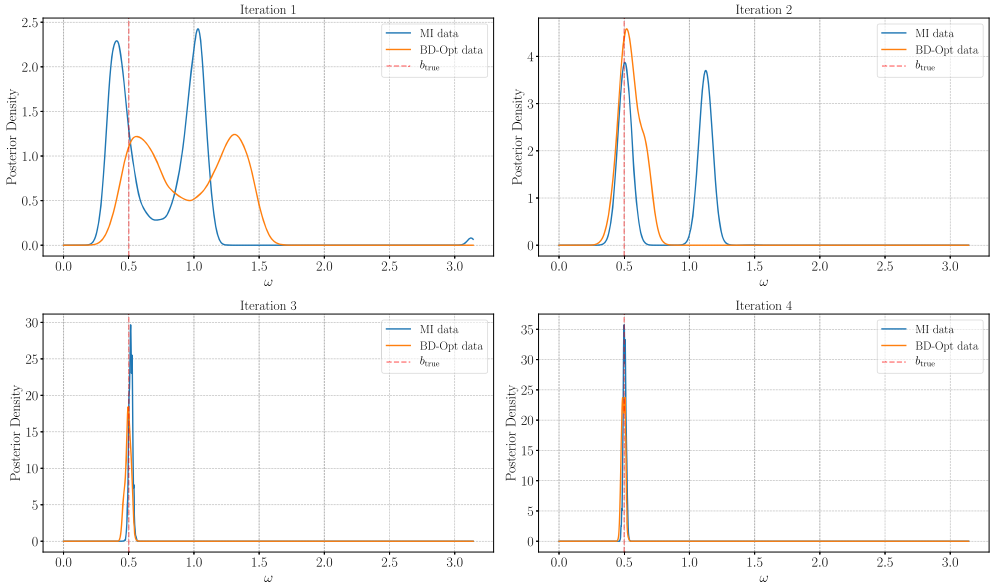

Figure 4: Comparison of the posterior distributions obtained when using the data from the MI and BD-Opt utilities for four iterations of the oscillatory toy model.

4.2 Death Model

The Death Model describes the stochastic decline of a population of N individuals due to some infection. Individuals change from the susceptible state S to the infected state I at an infection rate b, which is the model parameter we are trying to estimate. Each susceptible individual can get infected with a probability of $p_{\text{inf}}(t) = 1 - \exp(-bt)$ (Cook et al., 2008) at a particular time t. The aim of the Death Model is to decide at which measurement times τ to observe the infected population $I(\tau)$ in order to optimally estimate the true infection rate b.[7] Here we assume that for each iteration of the sequential BED scheme we only have access to a new, independent stochastic process. This means that, for instance, in iteration $k = 2$ we could have design times before the optimal design time τ_1^* of the first iteration.

The total number of individuals $\Delta I(t)$ moving from state S to state I at time t is given by a sample from a Binomial distribution,

$$\Delta I(t) \sim \text{Bin}(\Delta I(t); N - I(t), p_{\text{inf}}(\Delta t)), \tag{4.2}$$

where Δt is the step size, set to 0.01 in this work, and $I(t = 0) = 0$. By discretising this time series, the number of infected at time $t + \Delta t$ is given by $I(t + \Delta t) = I(t) + \Delta I(t)$. The likelihood for this model is analytically tractable (see Cook et al., 2008; Kleinegesse and Gutmann, 2019), and thus can serve as a means to validate our framework. As a

[7] See Appendix D for a time series plot of the Death Model.

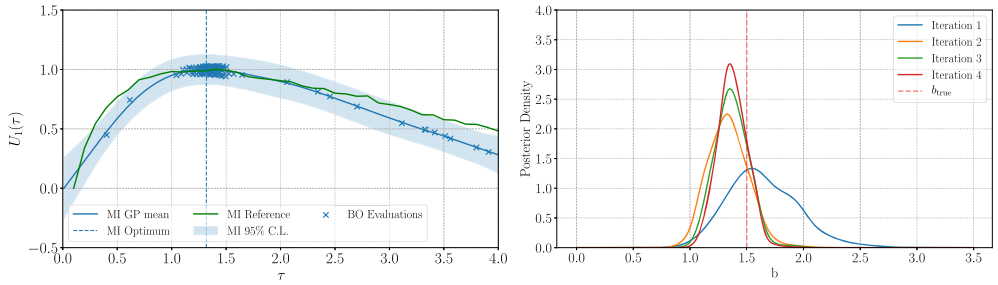

Figure 5: Left: MI utility for the first iteration of the Death model, including a reference MI computed numerically; shown are the GP mean and variance, BO evaluations and optimum of the GP mean. Right: Updated belief distributions after different iterations of the proposed sequential Bayesian experimental design approach for the Death model.

prior distribution we use a truncated Normal distribution, centred at 1 with a standard deviation of 1, while the summary statistics used to compute (2.11) are subsequent powers of the number of infected, i.e. $\boldsymbol{\psi}(I(\tau)) = [I(\tau), I(\tau)^2, I(\tau)^3]^\top$. To generate real-world observations, we use a true parameter value of $b_{\text{true}} = 1.5$.

We show the first iteration of the sequential mutual information utility in the left plot of Figure 5, as well as a reference MI value obtained using the tractable likelihood.[8] The MI peaks in the region around $\tau \approx 1$ and stays low at early and late measurement times. The average posterior for early and late τ is wider than the one for $\tau \approx 1$,[9] which results in a lower MI at the boundary regions. This is because at early and late measurement times the observed number of infected $I(\tau)$ is the same for a wide range of infection rates b, i.e. either 0 or 50 (the extreme values of $I(\tau)$). At $\tau \approx 1$ most values of b yield observations of $I(\tau)$ that are between the extreme values 0 and 50, allowing us to infer the relationship between b and $I(\tau)$ more effectively. For later iterations, the MI generally had the same form and did not change much, with optimal measurement times that were all roughly around 1 (see Appendix F for a plot with all iterations). This reduces the uncertainty in b which, in this case, outweighs the advantages of making an observation at different measurement times such as near the boundaries.

We show a KDE of the updated belief samples after each iteration, obtained by means of Algorithm 1, in the right plot of Figure 5. Even though the updated belief distribution after the first iteration has an expected value that is close to the true parameter, the corresponding credibility interval is wide. The belief distributions in the following iterations become more narrow, which is a result of having more data to estimate the model parameter. After four iterations, the posterior mean of b equals $\widehat{b} = 1.376$ with a 95% credibility interval of $[1.128, 1.621]$ containing $b_{\text{true}} = 1.5$. The credibility intervals were computed using a Gaussian KDE over posterior samples and the HPDI method.

[8] See Appendix E for a derivation of the reference MI.
[9] We show posterior plots for different measurement times in Appendix F.

4.3 SIR Model

The SIR Model (Allen, 2008) is an extension of the Death model and, in addition to the number of susceptibles $S(t)$ and infected $I(t)$, includes one more state population, the number of individuals $R(t)$ that have recovered from the infection and cannot be infected again. Similar to the Death model, the design variable is the measurement time τ at which to observe the state populations. For this model however, we are trying to estimate two model parameters, the rate of infection β and the rate of recovery γ. Similar to the Death model, we assume that for each iteration of the sequential BED scheme we only have access to a new, independent stochastic process.

At a particular time t of the time-series of state populations, let the number of individuals that get infected during an interval Δt, i.e. change from state $S(t)$ to state $I(t)$, be $\Delta I(t)$. Similarly, let the number of infected that change to the recovered state be $\Delta R(t)$. We compute these two state population changes by sampling from Binomial distributions,

$$\Delta I(t) \sim \text{Bin}(S(t), p_{\text{inf}}(t)) \tag{4.3}$$
$$\Delta R(t) \sim \text{Bin}(I(t), p_{\text{rec}}(t)), \tag{4.4}$$

where the probability $p_{\text{inf}}(t)$ of a susceptible getting infected is defined as $p_{\text{inf}}(t) = \beta I(t)/N$, where $\beta \in [0, 1]$ and N is the total (constant) number of individuals. The probability $p_{\text{rec}}(t)$ of an infected individual recovering from the disease is defined as $p_{\text{rec}}(t) = \gamma$, where $\gamma \in [0, 1]$. These state population changes define the unobserved time-series of the state populations S, I and R according to

$$S(t + \Delta t) = S(t) - \Delta I(t) \tag{4.5}$$
$$I(t + \Delta t) = I(t) + \Delta I(t) - \Delta R(t) \tag{4.6}$$
$$R(t + \Delta t) = R(t) + \Delta R(t) \tag{4.7}$$

We use initial conditions of $S(t = 0) = N - 1$, $I(t = 0) = 1$ and $R(t = 0) = 0$, where we set N to 50 and use a time-step of $\Delta t = 0.01$ throughout. The actual time at which we take observations is given by τ, such that the observed data is a single value for each state population, i.e. $S(\tau)$, $I(\tau)$ and $R(\tau)$. We use an uninformative, uniform prior $\mathcal{U}(0, 0.5)$ for both model parameters β and γ to draw initial prior samples. For the summary statistics used to compute (2.11) during the LFIRE algorithm we use subsequent powers, up to 3, of $I(\tau)$ and $R(\tau)$, including their products.[10]

The first four sequential MI utilities of the sequential BED scheme for the SIR model are shown in Figure 6. The SIR model utilities appear similar to those of the Death Model, with the main difference being that the global optima are shifted more towards lower measurement times around $\tau \approx 0.5$, increasing subtly with every iteration. Similar to the Death model, early and late measurement times result in posterior distributions that are, on average, wider than those for $\tau \approx 0.5$. This is because at early and late τ much of the data is the same for a wide range of model parameters $\boldsymbol{\theta}$. At early τ we mostly observe $S(\tau) = 49$, $I(\tau) = 1$ and $R(\tau) = 0$, i.e. the initial conditions. At late

[10] i.e. $\psi(\mathbf{y}) = [I(\tau), I(\tau)^2, I(\tau)^3, R(\tau), R(\tau)^2, R(\tau)^3, I(\tau)R(\tau), I(\tau)^2R(\tau), I(\tau)R(\tau)^2]^\top$.

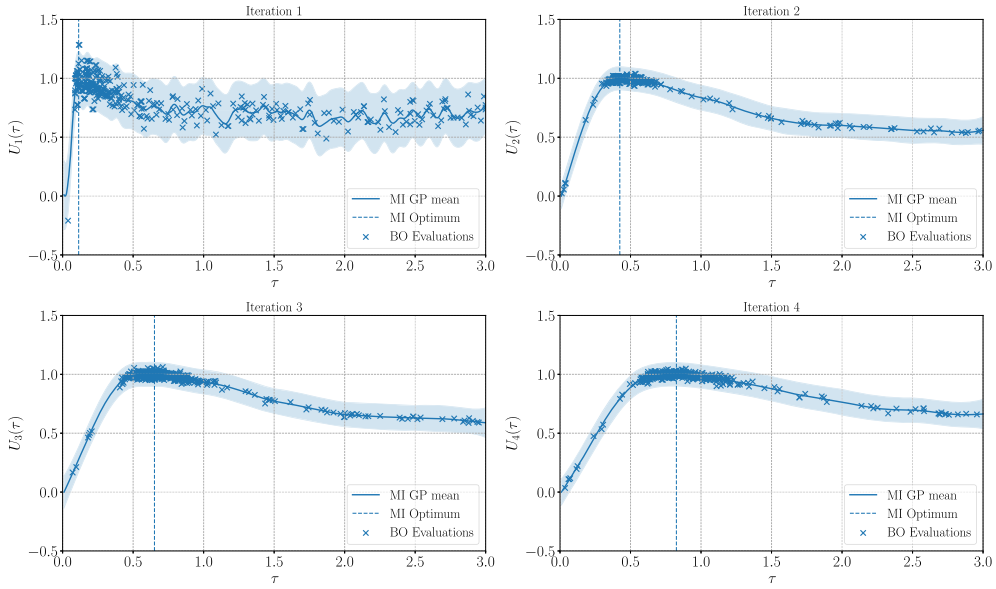

Figure 6: MI utilities at different iterations for the SIR model. Shown are the GP means and variances, BO evaluations and optima of the GP means.

measurement times there are no infected anymore, i.e. $I(\tau) = 0$, and we observe a fixed $S(\tau)$ and $R(\tau)$ that depend on the model parameters (see Appendix D for a typical time-series plot). Because the final values of $S(\tau)$ and $R(\tau)$ depend on the model parameters, late measurement times result in posteriors that are slightly more narrow than those for early measurement times. This is reflected in Figure 6, where the MI is higher at late τ than at early τ. At $\tau \approx 0.5$ we often have numbers of infected $I(\tau)$ that are non-zero, allowing us to infer the relationship between model parameters and data more effectively. This means that the resulting posterior for these measurement times is more narrow than elsewhere, where $I(\tau)$ is close to zero, which leads to the global MI maxima that we see in Figure 6.

Similar to the Death model, the form of the utilities for the SIR model does not change much between iterations. The utility for the first iteration appears noisier than the other ones because the parameter samples during that iteration stem from a uniform prior, which is highly uninformative and increases the Monte-Carlo error of the sample average in (3.4). The other utilities do not see this issue as the parameter samples from the updated belief distributions are less spaced out than for the first iteration. Resampling was performed according to Algorithm 2 during iterations 2–4, as the effective sample size always went below 50%.

The posterior densities after every iteration are shown in Figure 7 in form of KDEs computed from the posterior samples obtained according to Algorithm 1. We see that the beliefs about the model parameters become more precise after every iteration. This visualises that data acquired around measurement time $\tau \approx 0.5$ are providing useful

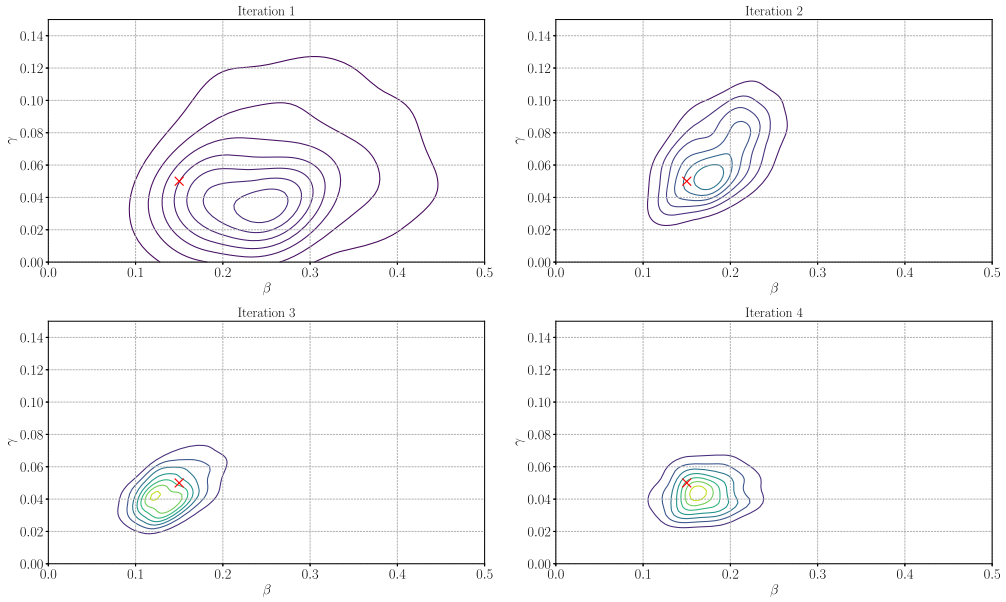

Figure 7: Comparison of the posterior densities at different iterations for the SIR model. The true model parameters are shown with a red cross.

information about the model parameters. After four iterations, the mean estimate of the infection rate β is $\widehat{\beta} = 0.171$ with a 95% credibility interval of $[0.112, 0.233]$. The corresponding mean estimate of the recovery rate γ is $\widehat{\gamma} = 0.045$ with a 95% credibility interval of $[0.024, 0.068]$. Similar to before, the credibility intervals were computed using a Gaussian KDE of the posterior samples and the HPDI method. The true parameters used to generate real-world observations were $\beta_{\text{true}} = 0.15$ and $\gamma_{\text{true}} = 0.05$, which are both contained in the credibility intervals.

Overall, the SIR model example illustrates that we can effectively use the MI utility, computed and optimised via Algorithm 3, to perform sequential BED for an implicit model, where the likelihood function is intractable.

4.4 Cell Model

The cell model (Vo et al., 2015) describes the collective spreading of cells on a scratch assay, driven by the motility and proliferation of individual cells, with particular applications in wound healing (e.g. Dale et al., 1994) and tumor growth (e.g. Swanson et al., 2003). In the context of our work, the experimental design is about deciding when to count the number of cells on the scratch assay in order to optimally estimate the cell diffusivity and proliferation rate. Price et al. (2018b) used the cell model before to compare the synthetic likelihood and approximate Bayesian computation likelihood-free inference approaches to estimate these model parameters. Importantly, in their work

they assumed that they had access to 144 images of a scratch assay and went on to estimate the model parameters given that an experimenter could analyse and quantify the cell spreading in all 144 images. We here wish to find out which of these images an experimenter should analyse if there is a limited experimental budget.

The (discrete) cell model starts with a grid of size 27×36 and 110 initial cells that are randomly placed in the upper part of the grid. This simulates wound healing, where a part of the tissue was scratched away due to an accident. At each discrete time step, every cell in the grid has a chance of moving to a neighbouring, empty grid position, which is given by the model diffusivity D. Similarly, at each discrete time every cell also has a chance to reproduce and spawn a new cell in a neighbouring, empty position, which is dictated by the model proliferation rate λ. While the model parameters of interest are the diffusivity D and proliferation rate λ, it is often easier to work with the probability of motility $P_m \in [0, 1]$ and probability of proliferation $P_p \in [0, 1]$.[11] For a particular combination of $\{P_m, P_p\}$ we can then simulate a time-series of grids where cells move around and reproduce.[12] In the context of BED, the discrete design variable is then the time at which to observe this grid and count the total number of cells. In reality, a human would have to physically count the number of cells under a microscope, which is time-consuming, and therefore we want to find the optimal times at which to have the experimenter make an observation. Similar to previous models, we here assume that we have access to a new, independent stochastic process for each sequential iteration.

We shall use 144 time steps as Vo et al. (2015) and Price et al. (2018b), which means that, including the initial grid, there are 145 grids in every time-series. For the summary statistics used to compute (2.11), we use the Hamming distance between a particular grid and the initial grid, as well as the total number of cells in a particular grid. The one-dimensional design variable is discrete and can take values between 1 and 145, i.e. $d \in \{1, \dots, 145\}$, while the summary statistic is two-dimensional. For the model parameters we use prior distributions $p(P_m) = \mathcal{U}(P_m; 0, 1)$ and $p(P_p) = \mathcal{U}(P_p; 0, 0.005)$; we choose the true model parameters to be $P_{m,\text{true}} = 0.35$ and $P_{p,\text{true}} = 0.001$ as Price et al. (2018b). Because the simulation time for the cell model is significantly more expensive than the previous models we have tested, we only use 300 initial prior samples during the sequential BED algorithm, which we run up to five iterations. We note that, while decreasing the computational resources needed, this may increase the Monte-Carlo error in (3.4) and the error in the LFIRE ratio estimate.

In the top row in Figure 8 we compare the sequential MI utilities for iteration 1 (left) and 5 (right) for the cell model; see Appendix G for a plot showing utilities for all iterations. Shown are the posterior predictive means and variances of the surrogate GP model, the BO evaluations and the respective optima. Because of the discrete domain, a normal GP tends to overfit this data and therefore we have used a one-layer deep GP (Damianou and Lawrence, 2013) as the surrogate model, which means that the GP hyper-parameters are modelled by GPs as well. For each iteration there seems to be some merit in taking observations at small and large designs but not for medium-large

[11] See how these parameters can be converted in Vo et al. (2015).
[12] See Appendix D for a plot showing the spreading of cells under this model.

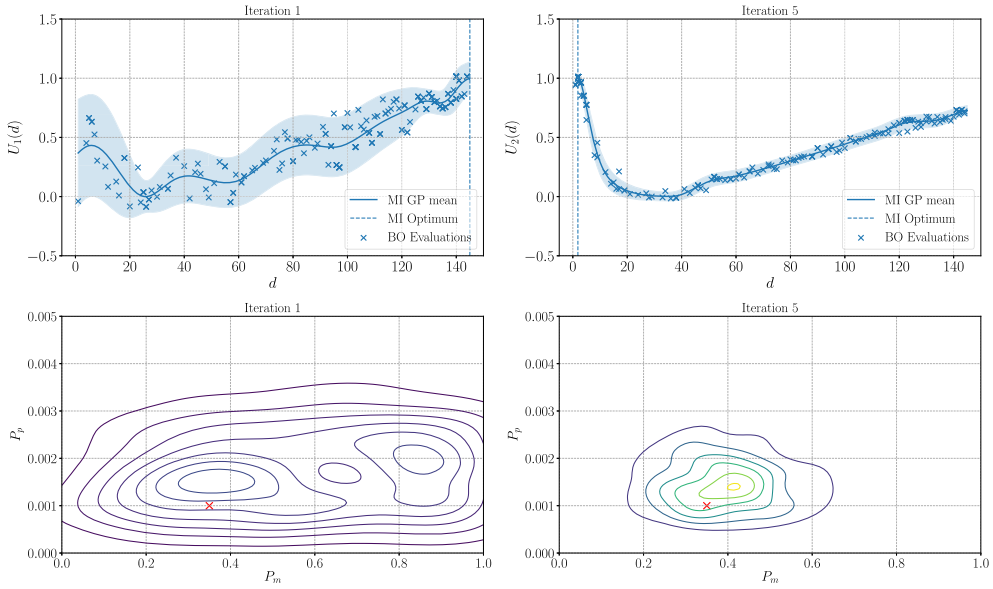

Figure 8: MI utilities (top row) and posteriors (bottom row) for iteration one and five for the cell model. The true model parameters are shown with a red cross.

designs, e.g. $d \sim 20$–40. At early designs, not much proliferation will have happened (due to its small prior probability) and so one can more easily measure the effect of motility. Conversely, at large designs one can average out the effect of motility and more easily notice the effect of proliferation, as more time has elapsed. In our case, the information gain about proliferation seems to beat that about motility in iteration 1, and similarly for iterations 2–4 (see Appendix G). After having repeatedly made measurements at large designs, at iteration 5 it becomes more effective to measure at small designs. This is because we have decreased the uncertainty in the proliferation parameter in iterations 1–4 and then need a measurement at early designs to sufficiently decrease the uncertainty in the motility parameter. Note that we resampled the parameters according to Algorithm 2 before iteration 2 and 3, as the effective sample size went below 50%.

Similarly to before, we use KDE and updated belief samples obtained from Algorithm 1 to visualise the approximate posterior densities after every iteration. In the bottom row in Figure 8 we show the posterior densities obtained after iteration 1 (left) and 5 (right); see Appendix G for a plot showing posterior distributions after every iteration. After iteration 1, the updated belief distribution has a wide spread in the P_m parameter and is more narrow for the P_p parameter. The optimal design at iteration 1 was at the far end of the design domain at $d_1^* = 145$. This is a design that helps to more easily detect the effect of proliferation as opposed to motility, which is reflected in the figure. The same phenomenon occurs in subsequent iterations 2–4, where the optimal designs are at the far end of the design domain (see Appendix G). This results in posterior distributions that are relatively narrow for P_p but wider for P_m. Taking a

measurement at the small design $d_5^* = 2$ in iteration 5, which allows us to more easily detect the effect of motility, reduces the uncertainty in the P_m parameter as well, as can be seen in the bottom right plot of Figure 8. The mode of the posterior distribution after iteration 5 is close to the true parameter value of $P_{m,\text{true}} = 0.35$ and $P_{p,\text{true}} = 0.001$. The estimated mean of the motility parameter is $\widehat{P}_m = 0.394$ with a 95% credibility interval of $[0.166, 0.642]$, while for the proliferation parameter it is $\widehat{P}_p = 0.00150$ with a 95% credibility interval of $[0.00055, 0.00265]$. Both credibility intervals contain the true parameter values. The credibility intervals were computed using a Gaussian KDE of the marginal posterior samples and the HPDI method.

The cell model demonstrates that we can effectively use MI in sequential BED when the forward simulations are expensive and the design domain is discrete. For this model, an experimenter might intuitively want to take observations at regular intervals but we have seen from the sequential utility functions that the expected information gain may then be sub-optimal. Sequential BED suggests that, when there is a limited budget, it is best to first take observations at high values in the design domain as the effect of proliferation dominates that of motility. Later, however, we should take observations at small designs to reduce the uncertainty in the motility parameter as well.

5 Conclusion

In this work we have presented a sequential Bayesian experimental design (BED) framework for implicit models, where the data-generating distribution is intractable but sampling from it is possible. Our framework uses the mutual information (MI) between model parameters and data as the utility function, which has not been done before in the context of sequential BED for implicit models due to computational difficulties. In particular, we showed how to obtain an estimate of the MI by density ratio estimation methods and then optimise it with Bayesian optimisation. To estimate the MI in subsequent iterations, we showed how to obtain updated belief samples by using a weighted particle approach and updating weights every iteration using the computed density ratios. We devised a resampling algorithm that yields new parameter samples whenever the effective sample size of these weights went below a minimum threshold. The framework can be used to produce sequential optimal experimental designs that can guide the data-gathering phase in a scientific experiment.

We first illustrated and explained our framework on a oscillatory toy model with multi-modal posteriors and then applied it to more challenging examples from epidemiology and cell spreading. For all examples we obtained optimal experimental designs that made intuitive sense and resulted in informative posterior distributions. For the oscillatory toy model, we also compared MI to Bayesian D-Optimality (BD-Opt), which has been used in sequential BED once before by Hainy et al. (2016). We found that, besides being less computationally expensive, MI usually led to different optimal designs than BD-Opt, due to the latter penalising multi-modality and only focusing on posterior precision.

While we have applied our framework to implicit models with low dimensionality, the theory is general and extends to models with high-dimensional designs as well, albeit

being more computationally intensive. Standard Bayesian optimisation, as used in this work, becomes, however, expensive and less effective in high dimensions. One would either have to utilise recent advances in high-dimensional Bayesian optimisation or look towards alternative gradient-free optimisation schemes, such as for example approximate coordinate exchange (Overstall and Woods, 2017).

Our approach leverages likelihood-free inference by density ratio estimation (Thomas et al., 2016). As discussed by Thomas et al. (2016), this is an inference framework rather than a single method. Since our approach can be combined with any method from that framework, we expect that advances made in that framework can be used to refine the approach proposed in this paper.

The MI utility represents the information gain of an experiment and is thus focused on obtaining accurate estimation results. However, it does not take the computational or financial cost of the different experimental designs into account. For that purpose one may want to maximise a normalised information gain instead, where we, for example, divide the MI by the estimated cost of running the experiment. In addition to prior knowledge, we may also use the costs from previous runs to predict the cost of the subsequent run in the sequential setting. Such approaches have been used in different contexts, for Bayesian optimisation of neural network hyperparameters (Snoek et al., 2012), but could also be useful here.

In our paper, we investigated experimental design for parameter estimation. However, we note that the proposed framework could also be applied to experimental design for model discrimination, as well as dual-purpose model discrimination and parameter estimation. Moreover, our paper focused on myopic sequential BED, where data is acquired one at a time and the experimental design does not explicitly take into account possible future experiments. It would be fruitful to investigate how our method applies to the case of non-myopic sequential BED, where we would plan ahead more than one experiment.

Supplementary Material

Supplementary Material of "Sequential Bayesian Experimental Design for Implicit Models via Mutual Information" (DOI: 10.1214/20-BA1225SUPP; .pdf). This includes Appendices A–G, which cover alternative estimation methods of the BD-Opt and sequential MI utility, more thorough explanations of parameter space transformations and reference calculations, as well as additional figures. The Python research code can be found as an installable package at `https://github.com/stevenkleinegesse/seqbed`, including examples relevant to this paper.

References

Agostinelli, S., Allison, J., Amako, K., Apostolakis, J., M Araujo, H., Arce, P., Asai, M., A Axen, D., Banerjee, S., Barrand, G., Behner, F., Bellagamba, L., Boudreau, J., Broglia, L., Brunengo, A., Chauvie, S., Chuma, J., Chytracek, R., Cooperman,

G., and Zschiesche, D. (2003). "GEANT4-a simulation toolkit." *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506: 250. 2

Allen, L. J. S. (2008). *Mathematical Epidemiology*, chapter An Introduction to Stochastic Epidemic Models, 81–130. Berlin, Heidelberg: Springer Berlin Heidelberg. MR2428373. doi: https://doi.org/10.1007/978-3-540-78911-6_3. 2, 12, 19

Alsing, J., Wandelt, B., and Feeney, S. (2018). "Massive optimal data compression and density estimation for scalable, likelihood-free inference in cosmology." *Monthly Notices of the Royal Astronomical Society*, 477: 2874–2885. 2

Arnold, B., Gutmann, M., Grad, Y., Sheppard, S., Corander, J., Lipsitch, M., and Hanage, W. (2018). "Weak Epistasis May Drive Adaptation in Recombining Bacteria." *Genetics*, 208(3): 1247–1260. 2

Bentley, J. L. (1975). "Multidimensional Binary Search Trees Used for Associative Searching." *Communications of the Association for Computing Machinery*, 18(9): 509–517. 10

Blum, M. and Francois, O. (2010). "Non-linear regression models for Approximate Bayesian Computation." *Statistics and Computing*, 20(1): 63–73. MR2578077. doi: https://doi.org/10.1007/s11222-009-9116-0. 6

Chen, Y. and Gutmann, M. U. (2019). "Adaptive Gaussian Copula ABC." In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, 1584–1592. 6

Chen, Z. (2003). "Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond." *Statistics*, 182. 9

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. 12, 17

Corander, J., Fraser, C., Gutmann, M., Arnold, B., Hanage, W., Bentley, S., Lipsitch, M., and Croucher, N. (2017). "Frequency-dependent selection in vaccine-associated pneumococcal population dynamics." *Nature Ecology & Evolution*, 1: 1950–1960. 2

Dale, P. D., Maini, P. K., and Sherratt, J. A. (1994). "Mathematical modeling of corneal epithelial wound healing." *Mathematical Biosciences*, 124(2): 127 – 147. 21

Damianou, A. and Lawrence, N. (2013). "Deep Gaussian Processes." In Carvalho, C. M. and Ravikumar, P. (eds.), *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, 207–215. Scottsdale, Arizona, USA: PMLR. 22

Diggle, P. J. and Gratton, R. J. (1984). "Monte Carlo Methods of Inference for Implicit Statistical Models." *Journal of the Royal Statistical Society. Series B (Methodological)*, 46(2): 193–227. URL http://www.jstor.org/stable/2345504 MR0781880. 2

Dinev, T. and Gutmann, M. U. (2018). "Dynamic Likelihood-free Inference via Ratio Estimation (DIRE)." *arXiv e-prints*, arXiv:1810.09899. MR3747571. doi: https://doi.org/10.1007/s11222-017-9738-6. 6

Doucet, A. and Johansen, A. (2009). "A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later." *Handbook of Nonlinear Filtering*, 12. MR2884612. 9

Drovandi, C. C., Pettitt, A. N., and Lee, A. (2015). "Bayesian indirect inference using a parametric auxiliary model." *Statistical Science 2015*, 30(1): 72–95. MR3317755. doi: https://doi.org/10.1214/14-STS498. 6

Greenberg, D., Nonnenmacher, M., and Macke, J. (2019). "Automatic Posterior Transformation for Likelihood-Free Inference." In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 2404–2414. Long Beach, California, USA: PMLR. 5

Gutmann, M. and Corander, J. (2016). "Bayesian optimization for likelihood-free inference of simulator-based statistical models." *Journal of Machine Learning Research*, 17(125): 1–47. MR3555016. 5

Gutmann, M., Dutta, R., Kaski, S., and Corander, J. (2018). "Likelihood-free inference via classification." *Statistics and Computing*, 28(2): 411–425. MR3747571. doi: https://doi.org/10.1007/s11222-017-9738-6. 6

Hainy, M., Drovandi, C. C., and McGree, J. (2016). "Likelihood-free extensions for Bayesian sequentially designed experiments." In Kunert, J., Muller, C. H., and Atkinson, A. C. (eds.), *11th International Workshop in Model-Oriented Design and Analysis (mODa 2016)*, 153–161. Hamminkeln, Germany: Springer. 2, 6, 13, 14, 15, 16, 24

Hainy, M., Müller, W., and Wagner, H. (2015). "Likelihood-free simulation-based optimal design with an application to spatial extremes." *Stochastic Environmental Research and Risk Assessment*, 30. 2

Ikonomov, B. and Gutmann, M. (2020). "Robust Optimisation Monte Carlo." In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*. 6

Järvenpää, M., Gutmann, M., Vehtari, A., and Marttinen, P. (2019). "Efficient acquisition rules for model-based approximate Bayesian computation." *Bayesian Analysis*, 14(2): 595–622. MR3934099. doi: https://doi.org/10.1214/18-BA1121. 5

Järvenpää, M., Gutmann, M. U., Vehtari, A., and Marttinen, P. (2020). "Parallel Gaussian process surrogate Bayesian inference with noisy likelihood evaluations." *Bayesian Analysis*, in press. 5

Kish, L. (1965). *Survey sampling*. Chichester: Wiley New York. 9

Kleinegesse, S., Drovandi, C., and Gutmann, M. U. (2020). "Supplementary Material of "Sequential Bayesian Experimental Design for Implicit Models via Mutual Information"." *Bayesian Analysis*. doi: https://doi.org/10.1214/20-BA1225SUPP. 3

Kleinegesse, S. and Gutmann, M. U. (2019). "Efficient Bayesian Experimental Design for Implicit Models." In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, 476–485. PMLR. 2, 6, 11, 17

Kullback, S. and Leibler, R. A. (1951). "On information and sufficiency." *Annals of Mathematical Statistics*, 22: 79–86. MR0039968. doi: https://doi.org/10.1214/aoms/1177729694. 4

Lindley, D. (1972). *Bayesian Statistics*. Society for Industrial and Applied Mathematics. MR0329081. 2

Lintusaari, J., Gutmann, M., Dutta, R., Kaski, S., and Corander, J. (2017). "Fundamentals and Recent Developments in Approximate Bayesian Computation." *Systematic Biology*, 66(1): e66–e82. 5

Lueckmann, J.-M., Gonçalves, P. J., Bassetto, G., Öcal, K., Nonnenmacher, M., and Macke, J. H. (2017). "Flexible Statistical Inference for Mechanistic Models of Neural Dynamics." In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 1289–1299. USA: Curran Associates Inc. 5

Marttinen, P., Croucher, N., Gutmann, M., Corander, J., and Hanage, W. (2015). "Recombination produces coherent bacterial species clusters in both core and accessory genomes." *Microbial Genomics*, 1(5). 2

Meeds, T. and Welling, M. (2015). "Optimization Monte Carlo: Efficient and Embarrassingly Parallel Likelihood-Free Inference." In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28*, 2080–2088. Curran Associates, Inc. 6

Mockus, J., Tiesis, V., and Zilinskas, A. (1978). *The application of Bayesian methods for seeking the extremum*, volume 2. MR0471305. 11

Müller, P. (1999). "Simulation-Based Optimal Design." *Bayesian Statistics*, 6: 459 – 474. MR1723509. 11

Numminen, E., Cheng, L., Gyllenberg, M., and Corander, J. (2013). "Estimating the Transmission Dynamics of Streptococcus pneumoniae from Strain Prevalence Data." *Biometrics*, 69(3): 748–757. MR3106603. doi: https://doi.org/10.1111/biom.12040. 2

Overstall, A. M. and Woods, D. C. (2017). "Bayesian Design of Experiments Using Approximate Coordinate Exchange." *Technometrics*, 59(4): 458–470. MR3740963. doi: https://doi.org/10.1080/00401706.2016.1251495. 2, 11, 25

Papamakarios, G. and Murray, I. (2016). "Fast ϵ-free Inference of Simulation Models with Bayesian Conditional Density Estimation." In Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29*, 1028–1036. Curran Associates, Inc. 5

Papamakarios, G., Sterratt, D., and Murray, I. (2019). "Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows." In Chaudhuri, K. and

Sugiyama, M. (eds.), *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, 837–848. PMLR. 6

Price, D. J., Bean, N. G., Ross, J. V., and Tuke, J. (2018a). "An induced natural selection heuristic for finding optimal Bayesian experimental designs." *Computational Statistics and Data Analysis*, 126: 112–124. MR3808393. doi: https://doi.org/10.1016/j.csda.2018.04.011. 11

Price, L. F., Drovandi, C. C., Lee, A., and Nott, D. J. (2018b). "Bayesian Synthetic Likelihood." *Journal of Computational and Graphical Statistics*, 27(1): 1–11. MR3788296. doi: https://doi.org/10.1080/10618600.2017.1302882. 6, 21, 22

Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., and Feldman, M. W. (1999). "Population growth of human Y chromosomes: a study of Y chromosome microsatellites." *Molecular Biology and Evolution*, 16(12): 1791–1798. 5

Ricker, W. E. (1954). "Stock and Recruitment." *Journal of the Fisheries Research Board of Canada*, 11(5): 559–623. 2

Rubin, D. B. (1984). "Bayesianly Justifiable and Relevant Frequency Calculations for the Applied Statistician." *Annals of Statistics*, 12(4): 1151–1172. MR0760681. doi: https://doi.org/10.1214/aos/1176346785. 5

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(1): 128–154. MR3491282. doi: https://doi.org/10.1111/insr.12107. 1, 2, 3, 4, 13

Schafer, M. C. and Freeman, P. (2012). "Likelihood-Free Inference in Cosmology: Potential for the Estimation of Luminosity Functions." In *Statistical Challenges in Modern Astronomy V*, volume 902 of *Lecture Notes in Statistics*, 3–19. MR3220168. 2

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and de Freitas, N. (2016). "Taking the Human Out of the Loop: A Review of Bayesian Optimization." *Proceedings of the IEEE*, 104(1): 148–175. 11

Sisson, S., Fan, Y., and Beaumont, M. (2018). *Handbook of Approximate Bayesian Computation*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press. MR3930567. 5

Sjöstrand, T., Mrenna, S., and Skands, P. (2008). "A brief introduction to PYTHIA 8.1." *Computer Physics Communications*, 178: 852–867. 2

Snoek, J., Larochelle, H., and Adams, R. P. (2012). "Practical Bayesian Optimization of Machine Learning Algorithms." In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems (NIPS)*, 2951–2959. 25

Swanson, K. R., Bridge, C., Murray, J., and Alvord, E. C. (2003). "Virtual and real brain tumors: using mathematical modeling to quantify glioma growth and invasion." *Journal of the Neurological Sciences*, 216(1): 1 – 10. 21

Thomas, O., Dutta, R., Corander, J., Kaski, S., and Gutmann, M. U. (2016). "Likelihood-free inference by ratio estimation." *arXiv e-prints*, arXiv:1611.10242. 2, 6, 10, 25

Vo, B. N., Drovandi, C. C., Pettitt, A. N., and Simpson, M. J. (2015). "Quantifying uncertainty in parameter estimates for stochastic models of collective cell spreading using approximate Bayesian computation." *Mathematical Biosciences*, 263: 133–142. MR3327999. doi: https://doi.org/10.1016/j.mbs.2015.02.010. 2, 12, 21, 22

Wilkinson, R. (2014). "Accelerating ABC methods using Gaussian processes." In Kaski, S. and Corander, J. (eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, 1015–1023. 6

Wood, S. N. (2010). "Statistical inference for noisy nonlinear ecological dynamic systems." *Nature*, 466: 1102. 2, 6

Acknowledgments

Steven Kleinegesse was supported in part by the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1) and the University of Edinburgh. Christopher Drovandi was supported by an Australian Research Council Discovery Project (DP200102101).

Supplementary Material of "Sequential Bayesian Experimental Design for Implicit Models via Mutual Information"

Steven Kleinegesse*, Christopher Drovandi† and Michael U. Gutmann*

Appendix A: Alternative Form of the BD-Opt Utility

We noticed certain numerical instabilities with the Bayesian D-Optimality (BD-Opt) utility as used by Hainy et al. (2016),

$$U(\mathbf{d}) = \mathbb{E}_{p(\mathbf{y}|\mathbf{d})}\left[\frac{1}{\det(\text{cov}(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{d}))}\right]. \tag{A.1}$$

These instabilities arise because inside the above expectation we are computing the inverse of the determinant of the posterior covariance. If the exact value of $\det(\text{cov}(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{d}))$ is small, approximating the expectation in A.1 with a standard sample-average may lead to extremely large $U(\mathbf{d})$ evaluations. Furthermore, for implicit models we cannot compute the determinant of the covariance exactly but have to approximate it with samples from the posterior distribution, obtained via a sequential Monte-Carlo approach (see Hainy et al., 2016)). Poor approximations of this quantity may also lead to large spikes in utility evaluations. We partly rectified this in our approach by taking the median instead of the mean in the sampling-based computation of the expectation.

Ultimately, these spikes in utility evaluations arise from an inherent instability in the BD-Opt utility. Although we have not tested it, we believe that a more stable form of the BD-Opt utility might be the following,

$$U_{\text{stable}}(\mathbf{d}) = -\mathbb{E}_{p(\mathbf{y}|\mathbf{d})}\left[\log\left\{\det(\text{cov}(\boldsymbol{\theta} \mid \mathbf{y}, \mathbf{d}))\right\}\right]. \tag{A.2}$$

The natural logarithm of the determinant of the posterior is additively proportional to the differential entropy of the multivariate normal distribution. Thus, similar to the previous BD-Opt, this utility works well for posterior distributions that a nearly Gaussian and fails for highly non-Gaussian posteriors, e.g. multi-modal distributions. Furthermore, by applying Jensen's inequality for concave functions, the utility $U_{\text{stable}}(\mathbf{d})$ in A.2 can be interpreted as a lower bound on the logarithm of $U(\mathbf{d})$ in A.1.

Appendix B: Utility Estimation with Weighted Samples

We could also approximate the sequential mutual information utility in Section 3.1 of the main text directly with weighted prior samples instead of using posterior samples,

*University of Edinburgh, steven.kleinegesse@ed.ac.uk, michael.gutmann@ed.ac.uk
†Queensland University of Technology, c.drovandi@qut.edu.au

i.e.

$$\widehat{U}_k(\mathbf{d}) = \frac{1}{N}\sum_{i=1}^{N} \log\left[\widehat{r}_k(\mathbf{d}, \mathbf{y}^{(i)}, \boldsymbol{\theta}^{(i)}, \mathbb{D}_{k-1})\right] w_{k-1}(\boldsymbol{\theta}^{(i)}; \mathbb{D}_{k-1}), \tag{B.1}$$

where $\mathbf{y}^{(i)} \sim p(\mathbf{y} \mid \mathbf{d}, \boldsymbol{\theta}^{(i)})$, $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$ and the weights w_{k-1} are given by

$$w_{k-1}(\boldsymbol{\theta}; \mathbb{D}_{k-1}) = \prod_{s=1}^{k-1} \widehat{r}_s(\mathbf{d}_s^*, \mathbf{y}_s^*, \boldsymbol{\theta}, \mathbb{D}_{s-1}), \tag{B.2}$$

with $\widehat{r}_1(\mathbf{d}_1^*, \mathbf{y}_1^*, \boldsymbol{\theta}, \mathbb{D}_0) = \widehat{r}_1(\mathbf{d}_1^*, \mathbf{y}_1^*, \boldsymbol{\theta})$ and $w_0(\boldsymbol{\theta}) = 1 \,\forall\, \boldsymbol{\theta}$.

In theory, the estimator in (B.1) has a lower variance than the estimator that we presented in the main text. In practice, however, we did not observe a significant difference but noticed that the estimator in (B.1) had longer computation times. Thus, we opted to use the sampling-based approach presented in Section 3.1 of the main text instead.

Appendix C: Parameter Space Transformation

We here describe in more detail how we transform the parameter samples $\boldsymbol{\theta}$ and boundary conditions $\mathcal{B}$ for the resampling procedure explained in Section 3.3 of the main text. Let $\theta_j^{(i)}$ be the jth element of the parameter sample $\boldsymbol{\theta}^{(i)}$. If the parameters θ_j in $\boldsymbol{\theta}$ have different scales, the KD-Tree algorithm produces nearest neighbours that underestimate, or overestimate, the standard deviation σ during the resampling procedure (see Section 3.3 of the main text). We found that we can overcome this and increase robustness by transforming all parameter samples such that their elements are bound between 0 and 1. Thus, for every element of every parameter sample we perform the transformation $\theta_j'^{(i)} \leftarrow \theta_j^{(i)}$ as follows,

$$\theta_j'^{(i)} = \frac{\theta_j^{(i)} - \theta_j^{\min}}{\theta_j^{\max} - \theta_j^{\min}}, \tag{C.1}$$

where $\theta_j^{\max}$ and $\theta_j^{\min}$ are the maximum and minimum, respectively, of the set of parameter samples $\{\theta_j^{(i)}\}_{i=1}^N$ for the jth element. Now consider boundary conditions of the form $\mathcal{B}_j = [\mathcal{B}_j^-, \mathcal{B}_j^+]^\top$ for the jth element of the parameter $\boldsymbol{\theta}$, where $\mathcal{B}_j^-$ and $\mathcal{B}_j^+$ are the lower and upper boundary, respectively. We assume that beyond these boundaries the prior probability $p(\theta_j)$ is zero and therefore we cannot resample beyond these boundaries. Using the same $\theta_j^{\max}$ and $\theta_j^{\min}$ as before, we transform the boundaries as well, i.e.

$$\mathcal{B}_j'^{-/+} = \frac{\mathcal{B}_j^{-/+} - \theta_j^{\min}}{\theta_j^{\max} - \theta_j^{\min}}. \tag{C.2}$$

In order to transform the resampled parameter samples back to the original parameter space, we simply have to invert (C.1) and obtain an expression for $\theta_j^{(i)}$.

We note that transformations of weighted samples do not change the weight of the samples. This means that if the set $\{w_k^{(i)}, \boldsymbol{\theta}^{(i)}\}_{i=1}^{i=N}$ represents $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$ in $\boldsymbol{\theta}$ space, $\{w_k^{(i)}, \boldsymbol{\theta}'^{(i)}\}_{i=1}^{i=N}$ represents $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$ in $\boldsymbol{\theta}'$ space. In the main text, we modelled $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$ as a truncated mixture of Gaussians. The law of transformations of random variables implies that $p(\boldsymbol{\theta} \mid \mathbb{D}_k)$ is also a mixture of Gaussians. This can be seen as follows: Let $\boldsymbol{\theta}' = \mathbf{A}\boldsymbol{\theta}$ where $\mathbf{A}$ is an invertible matrix (in our case, it is a diagonal matrix but this does not matter for the calculations). Denote $p(\boldsymbol{\theta}' \mid \mathbb{D}_k)$ by $p_{\boldsymbol{\theta}'}(\boldsymbol{\theta}' \mid \mathbb{D}_k)$. By the law of transformations of random variables, we have

$$p(\boldsymbol{\theta} \mid \mathbb{D}_k) = p_{\boldsymbol{\theta}'}(\mathbf{A}\boldsymbol{\theta} \mid \mathbb{D}_k)|\det \mathbf{A}| \tag{C.3}$$

$$\propto\!\!\propto \sum_{i=1}^{N} W_k^{(i)} \mathcal{N}(\mathbf{A}\boldsymbol{\theta}; \boldsymbol{\theta}'^{(i)}, \mathbb{I}\sigma^2) \mathbb{1}_{\mathcal{B}'}(\mathbf{A}\boldsymbol{\theta})|\det \mathbf{A}| \tag{C.4}$$

where we have inserted the mixture of Gaussians model from the main text. We can now combine $|\det \mathbf{A}|$ with $\mathcal{N}(\mathbf{A}\boldsymbol{\theta}; \boldsymbol{\theta}'^{(i)}, \mathbb{I}\sigma^2)$ to obtain a Gaussian in $\boldsymbol{\theta}$ space with transformed mean and variance parameter. Let $\mathbf{m}^{(i)} = \mathbf{A}^{-1}\boldsymbol{\theta}'^{(i)}$ so that $\boldsymbol{\theta}'^{(i)} = \mathbf{A}\mathbf{m}^{(i)}$ and denote the dimension of $\boldsymbol{\theta}$ by d.

$$\mathcal{N}(\mathbf{A}\boldsymbol{\theta}; \boldsymbol{\theta}'^{(i)}, \mathbb{I}\sigma^2)|\det \mathbf{A}| = \frac{|\det \mathbf{A}|}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\left(\mathbf{A}\boldsymbol{\theta} - \boldsymbol{\theta}'^{(i)}\right)^\top \left(\mathbf{A}\boldsymbol{\theta} - \boldsymbol{\theta}'^{(i)}\right)}{2\sigma^2}\right) \tag{C.5}$$

$$= \frac{|\det \mathbf{A}|}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\left(\boldsymbol{\theta} - \mathbf{m}^{(i)}\right)^\top \mathbf{A}^\top\mathbf{A} \left(\boldsymbol{\theta} - \mathbf{m}^{(i)}\right)}{2\sigma^2}\right) \tag{C.6}$$

$$= \frac{|\det \mathbf{A}|\frac{1}{\sigma^d}}{(2\pi)^{d/2}} \exp\left(-\frac{\left(\boldsymbol{\theta} - \mathbf{m}^{(i)}\right)^\top \frac{\mathbf{A}^\top\mathbf{A}}{\sigma^2} \left(\boldsymbol{\theta} - \mathbf{m}^{(i)}\right)}{2}\right) \tag{C.7}$$

This is the probability density function of a Gaussian with mean $\mathbf{m}^{(i)} = \mathbf{A}^{-1}\boldsymbol{\theta}'^{(i)}$ and covariance matrix $\boldsymbol{\Sigma} = \sigma^2(\mathbf{A}^\top\mathbf{A})^{-1}$. Using furthermore that $\mathbb{1}_{\mathcal{B}'}(\mathbf{A}\boldsymbol{\theta}) = \mathbb{1}_{\mathcal{B}}(\boldsymbol{\theta})$, we obtain

$$p(\boldsymbol{\theta} \mid \mathbb{D}_k) \propto\!\!\propto \sum_{i=1}^{N} W_k^{(i)} \mathcal{N}(\boldsymbol{\theta}; \mathbf{m}^{(i)}, \boldsymbol{\Sigma}) \mathbb{1}_{\mathcal{B}}(\boldsymbol{\theta}), \tag{C.8}$$

which proves the claim. Thus an alternative view of the resampling procedure outlined in the main text is that we fit a model of the form (C.8) to the weighted samples in $\boldsymbol{\theta}$ space where σ^2, which defines $\boldsymbol{\Sigma}$, is the only free parameter.

Appendix D: Simulation Plots for All Models

We show simulations of data as a function of time for all models considered in the main text in Figure 1 (oscillation toy model), Figure 2 (death model), Figure 3 (SIR model) and Figure 4 (cell model). For the oscillation toy model, death model and SIR model we show means and standard deviations computed from 1,000 simulations of time-series. For the cell model we show images of the spread of cells at different timesteps between 1 and 144. For each model, all responses were simulated using the corresponding true model parameters considered in the main text.

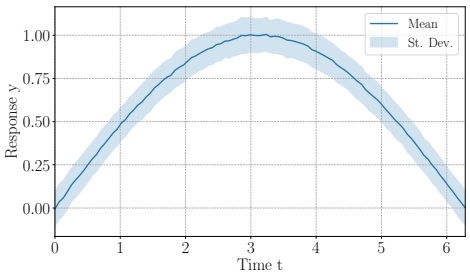

Figure 1: Sine model response as a function of time, computed with the true model parameters.

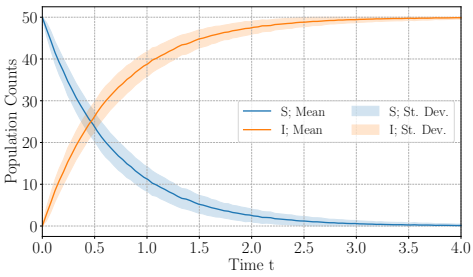

Figure 2: Death model population counts of S and I as a function of time, computed with the true model parameters.

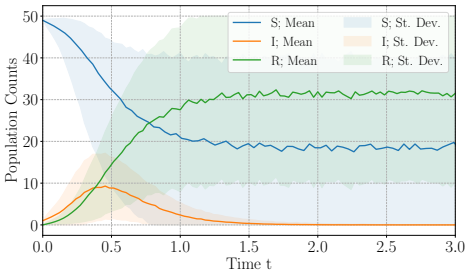

Figure 3: SIR model population counts of S, I and R as a function of time, computed with the true model parameters.

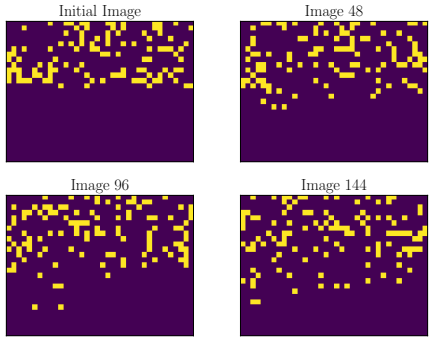

Figure 4: Cell model example simulation of cell motility and proliferation as a function of time, computed with the true model parameters.

Appendix E: Reference MI Computation

In order to compute reference mutual information (MI) values for the oscillation toy model and the death model, we use a nested Monte-Carlo sample-average. Note that we only compute reference MI values for the first sequential iteration. We assume that we can readily evaluate the data-generation distribution $p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})$ for these models and use this to compute a sample-average of the marginal data distribution, i.e. $p(\mathbf{y} \mid \mathbf{d}) \approx \frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y} \mid \boldsymbol{\theta}^{(j)}, \mathbf{d})$, where $\boldsymbol{\theta}^{(j)} \sim p(\boldsymbol{\theta})$. The mutual information is then approximated by

$$\mathrm{I}(\boldsymbol{\theta}; \mathbf{y} \mid \mathbf{d}) = \int p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d}) \log \left[\frac{p(\boldsymbol{\theta}, \mathbf{y} \mid \mathbf{d})}{p(\boldsymbol{\theta}) p(\mathbf{y} \mid \mathbf{d})} \right] \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{y} \tag{E.1}$$

$$= \int p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d}) p(\boldsymbol{\theta}) \log \left[\frac{p(\mathbf{y} \mid \boldsymbol{\theta}, \mathbf{d})}{p(\mathbf{y} \mid \mathbf{d})} \right] \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{y} \tag{E.2}$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} \log \left[\frac{p(\mathbf{y}^{(i)} \mid \boldsymbol{\theta}^{(i)}, \mathbf{d})}{\frac{1}{M} \sum_{j=1}^{M} p(\mathbf{y}^{(i)} \mid \boldsymbol{\theta}^{(j)}, \mathbf{d})} \right], \tag{E.3}$$

where $\mathbf{y}^{(i)} \sim p(\mathbf{y} \mid \mathbf{d}, \boldsymbol{\theta}^{(i)})$, $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$ and $\boldsymbol{\theta}^{(j)} \sim p(\boldsymbol{\theta})$.

For the sine model we use $p(y \mid \omega, t) = \mathcal{N}(y; \sin(\omega t), 0.1^2)$ in order to compute the reference MI according to (E.3) with $N = M = 1{,}000$. For the death model we use the data-generating distribution $p(S \mid b, \tau) = \mathrm{Bin}(S; S_0, \exp(-b(\tau - \tau_0)))$, where S is the number of susceptible individuals, $S_0 = 50$ and $\tau_0 = 0$ (Cook et al., 2008; Kleinegesse and Gutmann, 2019). The number of susceptibles can be computed from the number of infected individuals I by $S = S_0 - I$. We then compute the reference MI using (E.3) and $N = M = 1{,}000$.

Appendix F: Additional Plots for the Death Model

In Figure 5 we show average posterior densities at different measurement times for the death model. The posterior densities were approximated by using the analytic likelihood of the model and averaged over 100 observations $I(\tau^*)$ at $\tau^* \in \{0.1, 1.0, 4.0\}$.

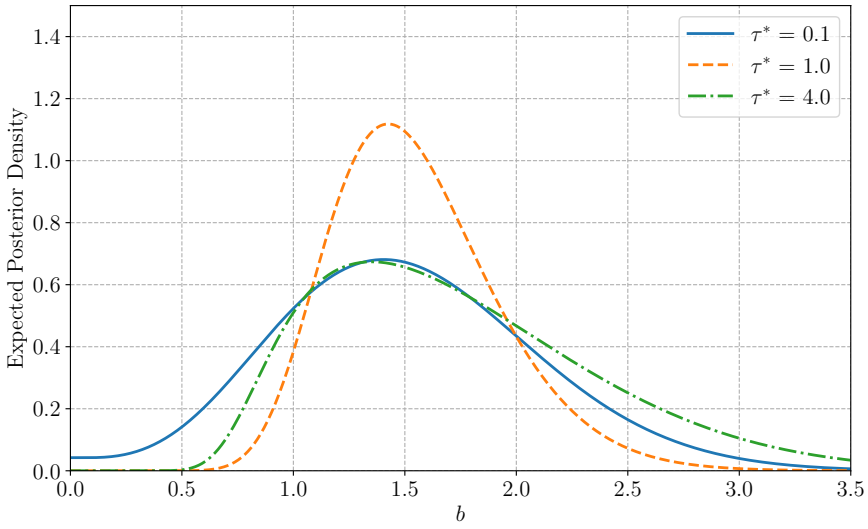

Figure 5: Expected posterior densities for the death model at different measurement times $\tau^* \in \{0.1, 1.0, 4.0\}$, averaged over 100 observations $I(\tau^*)$.

In Figure 6 we show the sequential MI utilities for all iterations of the death model, including the GP means and variances.

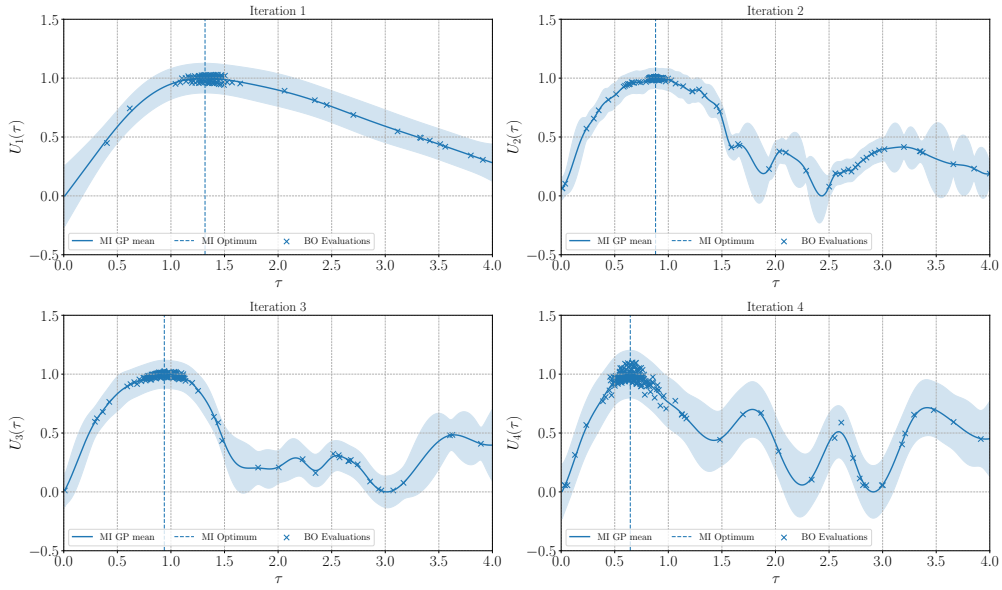

Figure 6: Sequential MI utilities for all iterations of the death model. Shown are the GP mean and variance, BO evaluations and optimum of the GP mean

Appendix G: Additional Plots for the Cell model

In Figure 7 we show the MI utilities for every iteration for the cell model. We show the corresponding posterior distributions after every iteration in Figure 8.

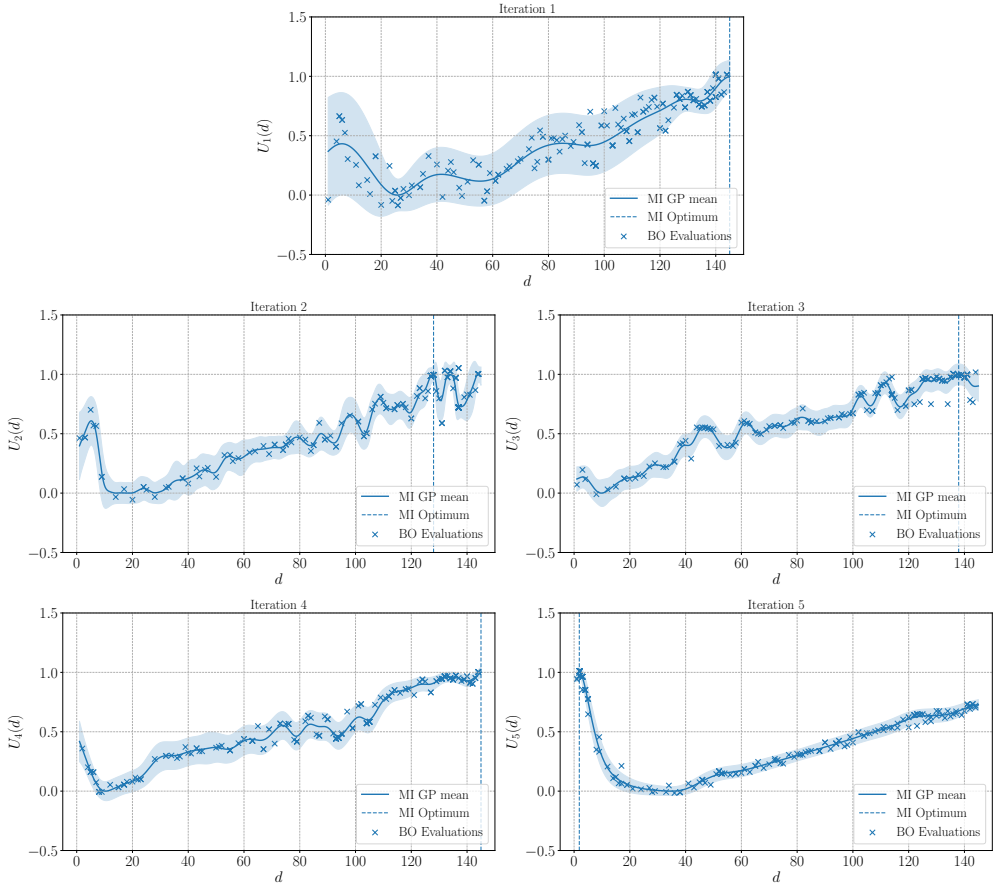

Figure 7: MI utilities at different iterations for the cell model. Shown are the GP means and variances, BO evaluations and optima of the GP means.

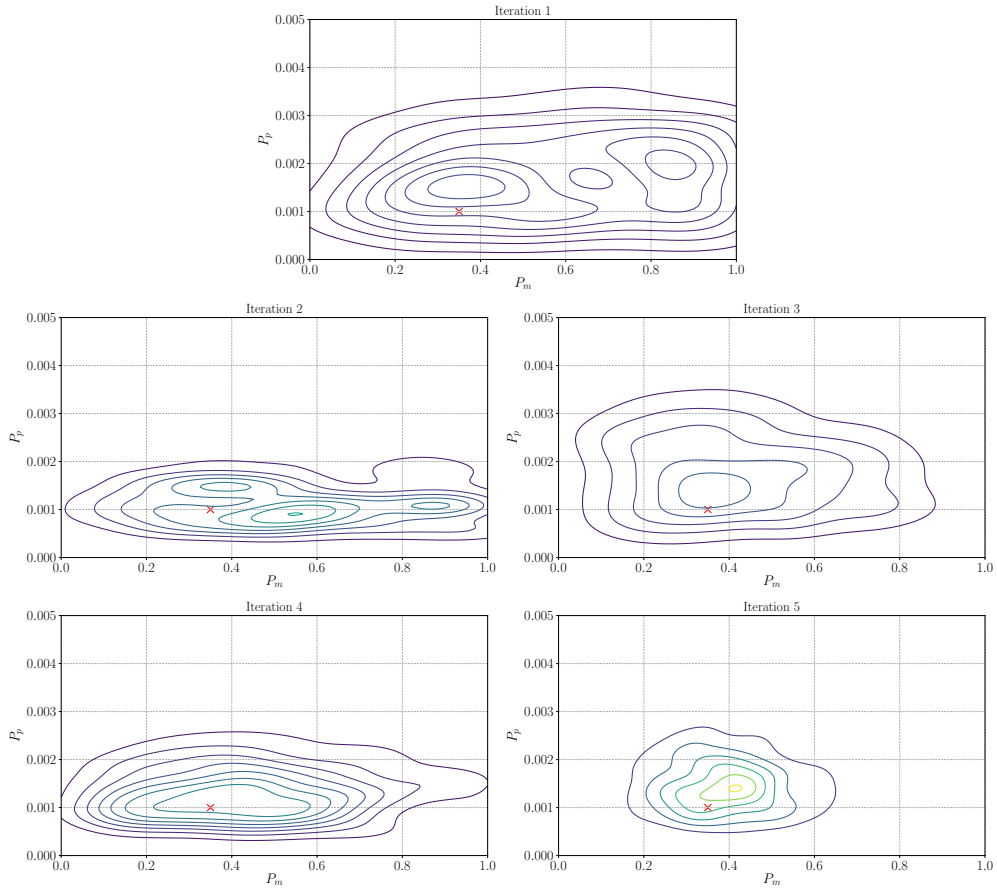

Figure 8: Comparison of updated belief distribution densities at different iterations for the cell model.

References

Cook, A. R., Gibson, G. J., and Gilligan, C. A. (2008). "Optimal Observation Times in Experimental Epidemic Processes." *Biometrics*, 64(3): 860–868. 5

Hainy, M., Drovandi, C. C., and McGree, J. (2016). "Likelihood-free extensions for Bayesian sequentially designed experiments." In Kunert, J., Muller, C. H., and Atkinson, A. C. (eds.), *11th International Workshop in Model-Oriented Design and Analysis (mODa 2016)*, 153–161. Hamminkeln, Germany: Springer. 1

Kleinegesse, S. and Gutmann, M. U. (2019). "Efficient Bayesian Experimental Design for Implicit Models." In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, 476–485. PMLR. 5