Journal Title
XX(X):1–24

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

# Efficient and flexible simulation-based sample size determination for clinical trials with multiple design parameters

**Duncan T. Wilson[1], Richard Hooper[2], Julia Brown[1], Amanda J. Farrin[1] and Rebecca E. A. Walwyn[1]**

**Abstract**

Simulation offers a simple and flexible way to estimate the power of a clinical trial when analytic formulae are not available. The computational burden of using simulation has, however, restricted its application to only the simplest of sample size determination problems, minimising a single parameter (the overall sample size) subject to power being above a target level. We describe a general framework for solving simulation-based sample size determination problems with several design parameters over which to optimise and several conflicting criteria to be minimised. The method is based on an established global optimisation algorithm widely used in the design and analysis of computer experiments, using a non-parametric regression model as an approximation of the true underlying power function. The method is flexible, can be used for almost any problem for which power can be estimated using simulation, and can be implemented using existing statistical software packages. We illustrate its application to a sample size determination problem involving complex clustering structures, two primary endpoints, and small sample considerations.

**Keywords**

Clinical trials, simulation, sample size, power, Gaussian process, global optimisation

[1]Leeds Institute of Clinical Trials Research, University of Leeds, Leeds, UK
[2]Centre for Primary Care & Public Health, Queen Mary University of London, London, UK

**Corresponding author:**
Duncan T. Wilson, Clinical Trials Research Unit, Leeds Institute of Clinical Trials Research, University of Leeds, Leeds, LS2 9JT, UK
Email: d.t.wilson@leeds.ac.uk

*Prepared using* sagej.cls *[Version: 2017/01/17 v1.20]*

## 1 Introduction

The sample size of a clinical trial is typically chosen with respect to its power, which is the probability of a statistically significant result conditional on the parameter of interest being equal to the Minimal Clinically Important Difference (MCID) [1]. Since power will generally increase with sample size, a nominal power threshold (often 80 or 90%) is set and the smallest sample size which satisfies this is selected. For many sample size determination (SSD) problems, power can be calculated using a simple mathematical formula and the optimisation problem can be solved in a timely manner. When complexity in the trial design or the method of analysis mean such formulae are not readily available, we can estimate power using a Monte Carlo (MC) approximation [2,3]. To do so, we simply simulate several hypothetical sets of trial data under the alternative hypothesis, analyse each of these, and calculate the proportion of analyses which reject the null hypothesis. The simplicity and flexibility of the simulation method has seen it used for a variety of statistical models and study designs, including problems involving hierarchical models [4,5], proportional hazards models [6], logistic regression models [7], individual patient data meta-analyses [8], patient enrolment models [9], stepped wedge designs [10,11], and cluster randomised crossover designs [12]. Although calculating MC estimates of power can be computationally demanding, these SSD problems remain feasible because, as optimisation problems, they are quite simple. In particular, optimisation takes place over a single parameter (the sample size), subject to a single constraint (power), and with respect to a single objective to be minimised (the sample size again).

SSD problems, particularly those found in trials of complex interventions, are not always this simple [13]. There may be several dimensions to the trials sample size or, more generally, several quantitative parameters which must be specified at the design stage and which influence the power of the trial. We will refer to these as *design parameters*. Several design parameters are common in, for example, trials with multilevel structures such as cluster randomised trials, where both the number of clusters and the number of participants in each cluster must be specified. Increasing the number of design parameters complicates the SSD problem by increasing the number of possible solutions to search. A second complication arises when there is more than one criterion we are interested in minimising, subject to the nominal power constraint. A cluster randomised trial will often have this property, as we would like to minimise both the total number of participants and the number of clusters. Given multiple conflicting objectives, there is no single 'optimum' solution but rather a range of solutions which offer different degrees of trade-off between the objectives. Seeking a set of good solutions, rather than a single optimum, further adds to the difficulty of the SSD problem.

Simple, simulation-based SSD problems can generally be solved in a feasible time with an exhaustive search, [14,15] though a bisection search, as proposed by Williams and colleagues and refined by Jung, [16,17] may be more efficient. A more sophisticated algorithm was implemented in the SimSam package, [5] written for Stata (Stata Corporation, College Station, Texas USA). Although there is no such general framework for solving complex SSD problems, a number of methods have been proposed for specific applications. In some cases these methods impose restrictions which reduce the complex SSD problem to a simple one in a single dimension. For example, approaches to simulation-based SSD for stepped wedge trials have focused on choosing one design parameter (such as the number of clusters randomised to each sequence) while keeping others (the number of participants per cluster and the number and arrangement of sequences)

fixed.[10,11]. In other applications, including meta-analysis[18], work has focussed on estimating power through simulation but has left the associated complex SSD problem unspecified. The complexity of SSD in multilevel designs has been addressed to some extent in the MLPowSim package[15], although this is limited to performing a simple grid search over two design variables (the number of clusters, and the cluster size). Adaptive designs are another area where simulation is often used to estimate operating characteristics, and where the SSD problem is complex due to the large number of design parameters needed to describe stopping rules. Where these problems have been addressed, optimisation has remained feasible through some special structure of the problem being identified and exploited. For example, optimal multi-arm multi-stage designs can be found when the primary endpoint is normally distributed, since one can draw a single large sample from a multivariate normal distribution prior to optimisation and use transformations of these samples over the course of the search[19]. Together with an implementation in C++, this leads to feasible computation times.

In theory, a general approach to solving any complex SSD problem would be through using benchmark multi-objective optimisation algorithms such as NSGA-II[20], robust implementations of which are freely available in statistical software such as R[21]. However, these so-called 'greedy' algorithms typically assume that evaluating any proposed solution to the problem is a very fast process, and consequently evaluate many thousands of solutions during the search. If these algorithms were applied to problems where evaluating solutions required computing an MC estimate of power, they would take an infeasibly long time to converge. Thus, if we are to extend simulation-based trial design to complex SSD problems, we require a more general framework employing more efficient optimisation algorithms.

Outwith the context of clinical trial design, a great deal of research has addressed optimisation problems where the evaluation of a solution is a computationally demanding, or *expensive*, operation[22,23]. One approach addresses the problem by substituting the expensive function with a mathematical approximation known as a *surrogate model*. The surrogate model is then used to make predictions about the true function for different values of design parameters, with these predictions informing which point should be evaluated next. The information obtained from this evaluation is used to update the surrogate model, thus improving the predictions available at the next iteration. One class of surrogate model is Gaussian process (GP) regression. Also known as Kriging and having its roots in geostatistics[24], GP models are spatial interpolators which are computationally tractable[25] and can be fitted using robust and freely available software[26]. A GP surrogate model provides not only a prediction of the true function value at any point, but also a measure of uncertainty in this prediction. This property is exploited by the benchmark Efficient Global Optimisation (EGO) algorithm[27], allowing the next point in the search to be chosen in a way that formally balances the potential benefits of *exploitation* (searching around areas already known to be promising) and *exploration* (searching in areas of high uncertainty). Although EGO was originally proposed for unconstrained optimisation of expensive objective functions with deterministic output, various proposals have extended it to incorporate the expensive constraints[28], multiple objectives[29], and stochastic outputs[30] that feature in complex SSD problems.

In this paper we will explore how GP regression models and a variant of the EGO algorithm can be used to solve complex SSD problems. We will describe a general method which can be

applied to almost any SSD problem, providing the user can write a program which simulates the data generating process and analysis of the trial at any given set of parameter values and for a given choice of sample size, returning a binary indicator denoting rejection or otherwise of the null hypothesis. By allowing user-written simulation programs, we aim to provide a flexible method which can be applied to a wide range of trial design problems. This approach will also facilitate SSD for novel designs developed in the future and which cannot be anticipated now.

The remainder of the paper is structured as follows. A motivating example is described in Section 2. In Section 3 we provide the necessary background and notation regarding Monte Carlo estimation and multi-objective optimisation. In Section 4 we describe Gaussian process regression, the efficient global optimisation algorithm, and a framework for its application to sample size determination. We return to the examples in Section 5, illustrating how the method can be applied in practice, where we also conduct a simulation study. We conclude with a discussion of the implications and limitations of the proposed approach in Section 6.

## 2 Motivating example - PACE

'Pacing, graded Activity, and Cognitive behaviour therapy; a randomised Evaluation' (PACE)[31,32] was a randomised controlled trial comparing adaptive pacing therapy, cognitive behavioural therapy, graded exercise therapy and specialist medical care as secondary care treatments for patients with chronic fatigue syndrome, with respect to two potentially correlated continuous primary endpoints (fatigue and disability). Participants were randomised to receive one of these four treatments, and we will henceforth refer to these randomised groups as 'arms' of the trial. The data had a complex multilevel structure, with three of the four arms including a therapy provided by different therapists (partially nested structure) and all arms including medical care from the same doctors (crossed structure), leading to the potential for treatment-related clustering. Here, we consider how we might have designed a pilot trial prior to the definitive PACE trial to provide a preliminary test of the potential efficacy of adaptive pacing therapy (APT) in comparison to Specialist Medical Care (SMC).

We assume the pilot trial would have the same complex multilevel data structure as the main trial, where therapists are partially nested within interventions, doctors are crossed with interventions, and patients are cross-classified with therapists and doctors in the intervention arm and nested within doctors in the control arm[33]. This structure is illustrated in Figure 1. As in the original PACE trial, the primary analysis will fit a mixed effect model for each primary endpoint and use likelihood ratio tests to obtain a p-value in each case. By fitting separate models for each endpoint, as opposed to a joint model, misspecification will be a potential problem if there are correlations between the endpoints at the participant level, or between the random effects of therapists and doctors. Such model misspecification has been noted as a clear motivation for simulation-based power calculations[3], which will allow an estimate of the true power, accounting for the effect of this misspecification, to be obtained.

The results of the trial will be considered positive if either of the analyses show a statistically significant difference, leading to an inflated type I error rate under the null hypothesis of no effect on each endpoint[34]. This inflation, together with the small sample context, motivates the inclusion of the nominal type I error rate $a$ as a design parameter in this problem. Together with

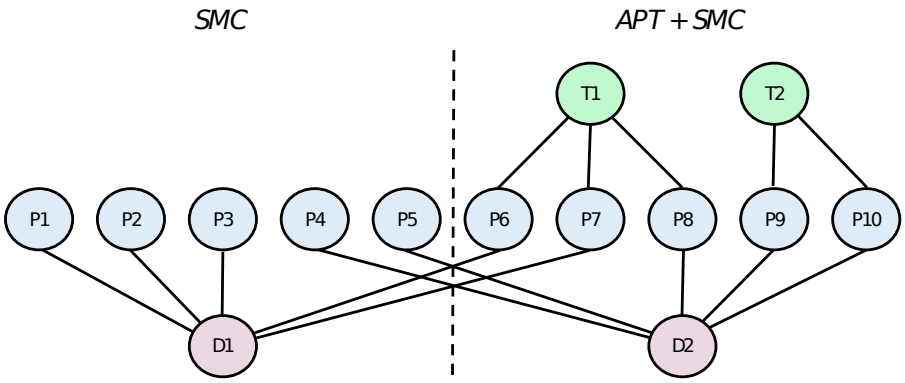


**Figure 1.** Multilevel structure of the SMC and APT arms of our example, where therapists (T) are partially nested within interventions, doctors (D) are crossed with interventions, and patients (P) are cross-classified with therapists and doctors in the intervention arm and nested within doctors in the control arm.

$a$, we have four further design parameters: the total sample size in the APT arm, denoted $n_1$; the number of APT therapists, $k$; the allocation ratio relating the total number of participants in each arm, $r = n_0/n_1$; and the number of doctors, $j$. Note that we assume we can control the numbers of participants allocated to therapists and to doctors, and so can maximise efficiency by balancing cluster sizes. The three objectives to be simultaneously minimised are the total number of participants, the number of therapists, and the number of doctors. Finally, we have the two constraints of ensuring the actual type I error rate is no more than 0.2 (one-sided), and the power is at least 90%. As both type I error rate and power do not have analytical forms in this context, both constraints must be estimated using simulation.

## 3 Background

### 3.1 Monte Carlo estimation

Monte Carlo methods can be used to numerically approximate expectations $\mathbb{E}[f(Z)]$ of real valued functions $f(Z)$ with respect to the probability distribution of $Z$. Given a sample of $N$ independent copies of $Z$, denoted $z_i$, $i = 1, \ldots, N$, the MC estimate is

$$\mathbb{E}[f(Z)] \approx \frac{1}{N} \sum_{i=1}^{N} f(z_i). \tag{1}$$

The estimate is unbiased for all $N$ and has variance equal to

$$\omega^2 = Var\left[\frac{1}{N} \sum_{i=1}^{N} f(z_i)\right] = \frac{1}{N} Var\left[f(z_i)\right]. \tag{2}$$

The standard error of the MC estimate will therefore reduce at a rate of $1/\sqrt{N}$ as we increase $N$. When $N$ is large we can consider an MC estimate to be the true expectation plus a normally

distributed error term with 0 mean and variance $\omega^2$, i.e.

$$\frac{1}{N}\sum_{i=1}^{N} f(z_i) = \mathbb{E}[f(Z)] + \epsilon, \text{ where } \epsilon \sim N(0, \omega^2). \tag{3}$$

In the context of simulation-based trial design, if $Z$ is the test statistic to be compared with an acceptance region $\Lambda$ then the probability of acceptance under hypothesis $H$ is $\mathbb{E}[I(Z \in \Lambda) \mid H]$, where $I(.)$ is the indicator function. An MC estimate of the power of a trial design under $H$ can therefore be obtained given $N$ test statistics $z_1, \ldots, z_N$ sampled under $H$. The steps required to simulate these statistics are described in [3], and we briefly summarise them here:

1. Define the population model. This describes the underlying target population and should specify all population parameters and distributions under the hypothesis of interest.
2. Define the sampling strategy. This should specify the numbers of patients, clusters, or any other sampling units in the trial and how they will be drawn from the population.
3. Define the method of analysis. For hypothesis testing, this will include defining the form of the test statistic $Z$ and the acceptance region $\Lambda$.

Given each of the above elements, pseudo-random number generators can be used to simulate the recruitment, randomisation and primary outcome measure of patients under the hypothesis of interest, from which a test statistic $z_i$ can be calculated.

### 3.2 Multi-objective optimisation

A solution to the SSD problem consists of a vector of design parameters $\mathbf{x}$. The *solution space* $\mathcal{X}$ is the set of all solutions. A simple SSD problem may have a 1-dimensional solution space, while more complex problems may have several dimensions. Elements of $\mathbf{x}$ may include parameters defining the sample size of the trial, the acceptance region to be used in the analysis, or any other design aspect over which we have control and which may influence the trial operating characteristics. For instance, in a cluster randomised trial we could define a solution by $\mathbf{x} = (k, n)$, where $k$ represents the number of clusters and $n$ the number of participants in each arm.

An *objective function* $f(\mathbf{x})$ is a function $f : \mathcal{X} \to \mathbb{R}$ which we wish to minimise. In a multi-objective problem with $B$ objectives, we denote the vector of objective values as $\mathbf{y} = (f_1(\mathbf{x}), \ldots, f_B(\mathbf{x})) \in \mathbb{R}^B$. We will describe $\mathbb{R}^B$ as the *objective space*. Extending the cluster randomised trial example, we have two objectives: minimising the number of clusters $f_1(\mathbf{x}) = k$; and minimising the total number of participants, $f_2(\mathbf{x}) = 2n$.

A *constraint function* $g(\mathbf{x})$ is a function $g : \mathcal{X} \to \mathbb{R}$ which must be less than or equal to 0 for the solution $\mathbf{x}$ to be considered feasible. For example, if type II error rate is denoted by $\beta(\mathbf{x})$ and the nominal type II error rate is set at $\beta^*$, a constraint function would be $g(\mathbf{x}) = \beta(\mathbf{x}) - \beta^*$. We denote $C$ constraint functions as $g_j(\mathbf{x}),\ j = 1, \ldots, C$. The general SSD problem can now be stated as

$$\min_{\mathbf{x} \in \mathcal{X}}\ f_i(\mathbf{x}),\ i = 1, \ldots, B \tag{4}$$

$$\text{subject to } g_j(\mathbf{x}) \leq 0,\ j = 1, \ldots, C. \tag{5}$$

We denote by $\prec$ the relation of Pareto dominance, where a solution dominates another if it is at least as good in all respects, and better in at least one. Formally, $\mathbf{x}_* \prec \mathbf{x}$ if $f_i(\mathbf{x}_*) \leq f_i(\mathbf{x})$ for $i = 1, \ldots, B$, and $f_j(\mathbf{x}_*) < f_j(\mathbf{x})$ for at least one $j$. For instance, $\mathbf{x}_a = (n = 100, k = 10) \prec \mathbf{x}_b = (n = 120, k = 10)$ in our cluster randomised trial example, but $\mathbf{x}_a = (n = 100, k = 10) \not\prec \mathbf{x}_c = (n = 80, k = 13)$. The *Pareto set* is the set of non-dominated solutions $\mathcal{X}_p = \{\mathbf{x} \in \mathcal{X} \mid \nexists\ \mathbf{x}_* \in \mathcal{X} \text{ s.t. } \mathbf{x}_* \prec \mathbf{x}\}$. An example Pareto set is plotted in Figure 2, illustrating the available trade-offs between the objectives of minimising the number of clusters and number of participants in a cluster trial.

Multi-objective optimisation seeks to find a set of solutions that are close to the true Pareto set, with every member of the set non-dominated with respect to all other members. We will refer to these as approximation sets, denoted $\mathcal{A}$. That is, any set $\mathcal{A} \subset \mathcal{X}$ such that $\{x \in \mathcal{A} : \exists\ \mathbf{x}_* \in \mathcal{A}, \mathbf{x}_* \prec \mathbf{x}\} = \emptyset$ is an approximation set [29]. A set $\mathcal{A}$ is feasible if all constraints are satisfied by every member of $\mathcal{A}$. An example feasible approximation set, plotted in Figure 2, is given by the four $(2n, k)$ points

$$\mathcal{A} = \{(589, 24), (705, 20), (810, 12), (982, 10)\}. \tag{6}$$

To understand the similarity between any approximation set $\mathcal{A}$ and the ideal Pareto set, we measure its dominated hypervolume. This is the volume of the subspace dominated by solutions in $\mathcal{A}$ and bounded by a reference point $r$:

$$H(\mathcal{A}) = \text{Vol}(\{\mathbf{y} \in \mathbb{R}^B \mid \mathbf{y} \text{ is dominated by some } \mathbf{y}_* \in \mathcal{A} \text{ and } \mathbf{y} \prec r\}). \tag{7}$$

The specific choice of $r$ is not important, providing it is larger than anticipated worst-case objective values. The largest possible hypervolume of any feasible approximation set $\mathcal{A}$ is achieved by the true Pareto set $\mathcal{X}_p$. We can therefore frame the multi-objective optimisation problem as finding the feasible approximation set $\mathcal{A}$ with largest hypervolume. Taking a reference point of $r = (1200, 30)$ (marked by the cross in Figure 2), our example approximation set has a dominated hypervolume of 9202. This can be compared with that of the Pareto set, at 14501. We would expect the approximation set to converge to the Pareto set as the number of optimisation iterations increases.

## 4 Simulation-based sample size determination

### 4.1 Overview

The proposed method is based on the Efficient Global Optimisation algorithm [35]. For clarity we will describe the algorithm in the context of an SSD problem with a single constraint function, denoted $g(\mathbf{x})$, which must be estimated using simulation. The more general case of several constraints will follow. The initial step is to select a number of potential solutions to the SSD problem $\mathcal{X}_E = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(E)})$ and evaluate the constraint function at each of these points, giving $\mathbf{y}_E = (g(\mathbf{x}^{(1)}), \ldots, g(\mathbf{x}^{(E)}))$. A Gaussian process regression model is then fitted to the data, relating the solutions $\mathcal{X}_E$ to the estimates $\mathbf{y}_E$ and providing an approximation of the constraint function $g$. The solution $\mathbf{x}_*$ which has the largest expected improvement $EI(\mathbf{x})$ according to the predictions of the GP model, is then found. This solution is evaluated to obtain

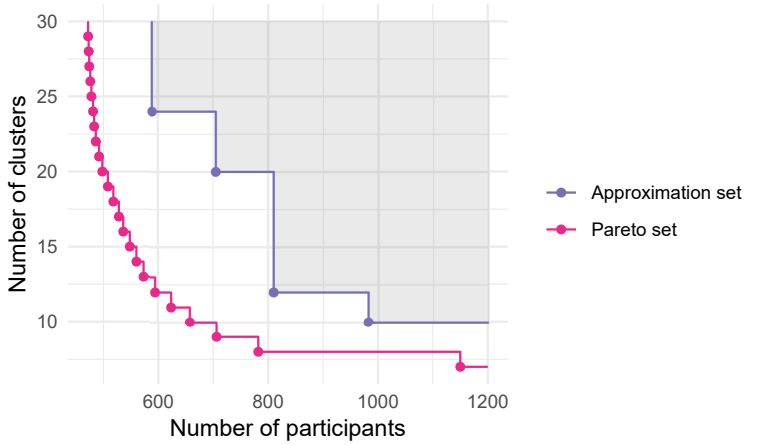


**Figure 2.** Example Pareto front $\mathcal{X}_p$ and approximation set $\mathcal{A}$ for a cluster randomised trial design problem. The dominated hypervolume of the approximation set with respect to a reference point (cross) is the shaded area.

$y_*$. This new data is then used to update the GP model, which is then used again to find the next solution to evaluate. The algorithm can be repeated until either the computational resources available have been exceeded, or until no further improvements are being obtained. The Algorithm is summarised in 1 below.

**Algorithm 1** Efficient Global Optimisation[35]

1: Compute MC estimates $\mathbf{y}_E = (g(\mathbf{x}^{(1)}), \ldots, g(\mathbf{x}^{(E)}))$
2: **while** Computation budget not exhausted **do**
3: Regress $\mathbf{y_E}$ on $\mathcal{X}_E = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(E)})$
4: Find $\mathbf{x}_* = \arg\max EI(\mathbf{x})$
5: Compute MC estimate $y_* = g(\mathbf{x}_*)$ and add to $\mathbf{y_E}$, $\mathcal{X}_E$
6: Update the computational budget
7: **end while**

The process of computing MC estimates used in steps (1) and (5) has already been described in Section 3.1. In what follows we will first consider step (3), describing Gaussian process regression models and outlining how they can be fitted and used to make predictions. The notion of expected improvement in step (4) will then be defined for the constrained multi-objective problems we are concerned with. Finally, we cover the remaining aspects of implementation.

## 4.2 Gaussian process regression

Consider a set of points $\mathcal{X}_E = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(E)}\} \subset \mathcal{X}$ at which an expensive function $g$ will be estimated using the Monte Carlo method. Consider also some other point $\mathbf{x}_* \notin \mathcal{X}_E$ where we are

interested in making a prediction of $g(\mathbf{x}_*)$. The value of $g$ at each point in $\{\mathcal{X}_E, \mathbf{x}_*\}$ is initially unknown, but can be modelled by a Gaussian process (GP).

In using a GP we assume that our belief regarding the the values of $g$ can be represented by a multivariate normal distribution. Prior to computing any estimates of $g$, we assume that the mean function of this multivariate normal is equal to zero*. We write the covariance matrix of the distribution as

$$\begin{pmatrix} K(\mathcal{X}_E, \mathcal{X}_E) & K(\mathcal{X}_E, \mathbf{x}_*) \\ K(\mathbf{x}_*, \mathcal{X}_E) & K(\mathbf{x}_*, \mathbf{x}_*) \end{pmatrix}, \tag{8}$$

where $K(\mathcal{X}_E, \mathcal{X}_E)$ is the $E \times E$ covariance matrix for the points $\mathcal{X}_E$, $\mathbf{k}_* = K(\mathcal{X}_E, \mathbf{x}_*) = K(\mathbf{x}_*, \mathcal{X}_E)$ is the $E$-length vector of covariances between $\mathcal{X}_E$ and $\mathbf{x}_*$, and $K(\mathbf{x}_*, \mathbf{x}_*)$ is the variance at $\mathbf{x}_*$.

Given this prior distribution, we compute the MC estimates $y^{(1)}, \ldots, y^{(E)}$ at each point in $\mathcal{X}_E$. From equation (3), $y^{(i)} = g(\mathbf{x}^{(i)}) + \epsilon^{(i)}$ where $e^{(i)}$ is a zero-mean normally distributed error term with standard deviation $\omega^{(i)}$. We denote by $\Delta$ the $E \times E$ diagonal matrix where the $i$th entry is $[\omega^{(i)}]^2$. The distribution of $g(\mathbf{x}_*)$ conditional on the observed $\mathbf{y}$ can be shown to be normal with mean $\mathbf{k}_*^\top (K + \Delta)^{-1}\mathbf{y}$ and variance $k(x_*, x_*) - \mathbf{k}_*^\top (K + \Delta)^{-1}\mathbf{k}_*$ [25]. Thus, given a prior covariance matrix of the form (8) and some MC estimates of $g$ at the points $\mathcal{X}_E$, a conditional predictive distribution of $g(\mathbf{x}_*)$ can be found. It is this distribution which will be used in the optimisation algorithm when deciding which solution should next be evaluated.

The predictive distributions are influenced by the prior covariance matrix (8). The matrix is populated using a covariance function (or *kernel*), $k(\mathbf{x}, \mathbf{x}') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. This function must be symmetric and positive definite for the covariance matrix to have the same properties. One such covariance function is the squared exponential, which has the form

$$k(\mathbf{x}, \mathbf{x}') = \sigma \exp\left( -\sum_{j=1}^{D} \frac{(x_j - x'_j)^2}{\lambda_j^2} \right). \tag{9}$$

By using covariance functions of this form we will obtain a Gaussian process which is infinitely differentiable over $\mathcal{X}$ and thus very smooth. This would appear to be a reasonable restriction to place upon the power functions we are interested in. In order to populate the covariance matrix we must choose values of the hyper-parameters $\boldsymbol{\theta} = (\sigma, \lambda_1, \ldots, \lambda_D)$. We do this by numerically optimising the log marginal likelihood

$$\log p(\mathbf{y} \mid \mathcal{X}_E, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^\top [K + \Delta]^{-1}\mathbf{y} - \frac{1}{2}\log|K + \Delta| - \frac{n}{2}\log 2\pi, \tag{10}$$

considered as a function of $\boldsymbol{\theta}$ [25]. Fitting a GP model by maximum likelihood in this manner can be done using the function `km` in the R package DiceKriging, as illustrated in the appendix.

An illustration of a Gaussian process regression model of a power function in one dimension is given in Figure 3. The power of three different choices of sample size have been calculated

*This is not a restrictive assumption. After observing estimates of the function $g$ and updating the GP model to account for these, the mean function can take on non-zero values.

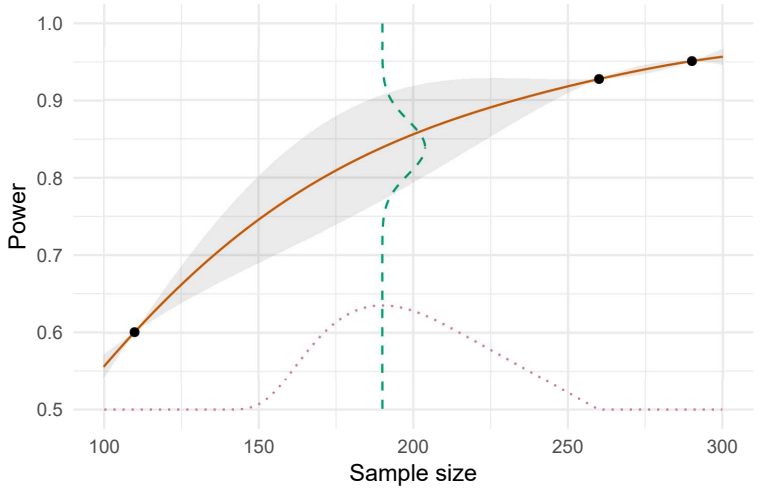


**Figure 3.** A Gaussian process model of a power function over a one-dimensional sample size (solid line) based on three evaluations. Uncertainty is shown as the shaded area. Expected improvement (dotted line) is maximised at a sample size of 190, where the predicted power is normally distributed around a mean estimate of 0.84 (dashed line).

and a GP model fitted to the results. The figure illustrates how the uncertainty in the model predictions (shaded area) increases the further we are from a point which has been evaluated. The GP prediction of power at a sample size of $n = 190$, shown as a dashed line, is normally distributed with mean 0.84 and standard deviation 0.035.

### *4.3 Expected improvement*

At any given point during the optimisation process we can obtain an approximation set $\mathcal{A}$ based on the set of solutions which have been evaluated up to that point. If a new point $\mathbf{x}_*$ is considered feasible, a new approximation set $\mathcal{A}_*$ will be identified. The improvement resulting from the evaluation of $\mathbf{x}_*$ is the difference in the dominated hypervolumes:

$$I = H(\mathcal{A}_*) - H(\mathcal{A}). \tag{11}$$

Prior to evaluation, we do not know if the point $\mathbf{x}_*$ will be considered feasible. We therefore modify $I$ to account for the probability that $\mathbf{x}_*$ will be considered feasible after the MC estimates have been obtained. This probability can be estimated using the GP regression methodology described in Section 4.2. A GP model of unknown constraint function $g$ will describe our current belief about the value of $g$ at $\mathbf{x}_*$ using a normal distribution with mean $m$ and variance $s^2$ $g(\mathbf{x}_*) \sim \mathcal{N}(m, s^2)$, and we will consider the point $\mathbf{x}_*$ feasible if the upper $100 \times p\%$ quantile of this distribution is below 0. We denote this quantile as

$$q(\mathbf{x}_*) = m + \Phi^{-1}(p)s, \tag{12}$$

where $\Phi$ is the standard normal cumulative distribution. Following an evaluation of $\mathbf{x}_*$ the GP model will be updated and the quantile revised to $q_+(\mathbf{x}_*)$. Before the evaluation the value of $q_+(\mathbf{x}_*)$ is unknown, but it is shown in[30] that its predictive distribution is $q_+(\mathbf{x}_*) \sim N(m_+, s_+^2)$ where

$$m_+ = m + \Phi^{-1}(p)\sqrt{\frac{\omega^2 s^2}{\omega^2 + s^2}} \tag{13}$$

$$s_+^2 = \frac{[s^2]^2}{\omega^2 + s^2}, \tag{14}$$

and $\omega$ is the MC error of the planned evaluation, estimated as $m(1-m)/N$ where $N$ is the number of MC samples to be used. The predictive distribution can then be used to calculate the probability that the point $\mathbf{x}_*$ will be considered feasible following its evaluation. Following[28], we multiply the theoretical improvement $I$ by this probability, thus penalising candidate solutions with a low chance of satisfying the constraint. This then gives us our expected improvement measure $EI$, where

$$\text{Expected Improvement } EI(\mathbf{x}_*) = [H(\mathcal{A}_*) - H(\mathcal{A})] \prod_{j=1}^{C} \Phi\left(\frac{-m_{j,+}}{s_{j,+}}\right). \tag{15}$$

Note that we include a penalty term for all $j = 1, \ldots, C$ constraint functions. This maximisation problem is in itself complex, with a potentially large number of local maxima. We therefore use the particle swarm optimisation algorithm as implemented in the R package pso[36], designed to avoid becoming trapped in local maxima, to solve this sub-problem.

An illustration of expected improvement for a single-objective problem is given in Figure 3. When choosing which sample size to evaluate next and aiming to find the lowest per-arm sample size with at least 80% power, we balance the potential improvement over the best current solution (a sample size of 260) with the probability of constraint satisfaction. In this case, we would choose to evaluate the sample size of 190, estimated by the GP model to have a power of 84%.

### 4.4 Fixed space-filling design

A alternative to the EGO algorithm is to evaluate a number of solutions chosen using a space-filling experimental design, and use these to construct an approximation set. Specifically, we construct a Sobol sequence (a low-discrepancy sequence which can be thought of as a quasi-random uniform distribution) and estimate the type II error rate at each of these solutions using $N$ MC samples. For each point a confidence interval based on the MC error can then be calculated, and any points where the interval was not entirely below the nominal value discarded. Of those that remained, any dominated solutions are discarded to leave an approximation set. We will refer to this as the 'fixed design' method.

### 4.5 Implementation

To apply Algorithm 1 in practice we must first choose the initial set of points to be evaluated, $\mathcal{X}_E$. One recommendation is to use a space-filling design with 10 points for each dimension of

the solution space, and to allocate between 30 and 50% of the total computation budget to their evaluation[37]. To select the location of the points in $\mathcal{X}_E$ we use the space-filling Sobol sequence generated using the R package randtoolbox[38]. The number of iterations and the number of MC samples $N$ used at each iteration must also be chosen. Given a total computational budget in terms of MC samples, the choice of these values should account for the fact that fitting GP regression models in R to more than around 800 points is currently infeasible[39]. The choice of the computational budget itself is not critical since, after terminating the algorithm, it can be simply restarted from its final state. This may be sensible if, for example, the trajectory of solution quality does not appear to have plateaued at termination.

As the algorithm depends on GP regression models, it can be helpful to assess the fit of these models. One approach is to regularly plot the predicted mean and standard deviation in one or two dimensions, centred at the last evaluated point. Poor model fit could be identified if the mean function is not, for example, strictly increasing as expected. We can also contrast the predicted function values with the obtained function values at each iteration, halting the algorithm if a large and unexpected discrepancy in these values is observed.

We have implemented the proposed framework in R, partly due to the availability of robust and efficient R packages for fitting Gaussian process models (DiceKriging[26]) and for global optimisation (pso[36]). Using R also provides flexibility in terms of the user-written simulation routines by facilitating various complicated analysis procedures, e.g. multilevel modelling through lme4[40]. The implementation is provided in the supplementary material and online at `https://github.com/DTWilson/Bayes_opt_SSD`, where we show how it can be applied to solve a number of example SSD problems.

## 5 Illustration and evaluation

### 5.1 Application to a hypothetical example

We begin by showing how the proposed method can be applied to design a hypothetical two-arm cluster randomised trial with a continuous, normally distributed outcome, $k$ clusters in each arm, and $m$ participants in each cluster. The outcome of participant $i$ in cluster $j$ is modelled as

$$y_{ij} = \beta_0 + \beta_1 t_i + u_j + e_i \tag{16}$$

$$u_j \sim N(0, \sigma_B^2) \tag{17}$$

$$e_i \sim N(0, \sigma_W^2), \tag{18}$$

where $t_i = 1$ if patient $i$ is in the intervention arm, and $t_i = 0$ otherwise. The trial will be analysed using a two-sample $t$-test comparing the cluster sample means

$$\bar{y}_j = \frac{1}{m} \sum_{i=1}^{m} y_{ij},$$

in the two arms. We assume the nuisance parameters are known and equal to $\sigma_W^2 = 0.95, \sigma_B^2 = 0.05$.

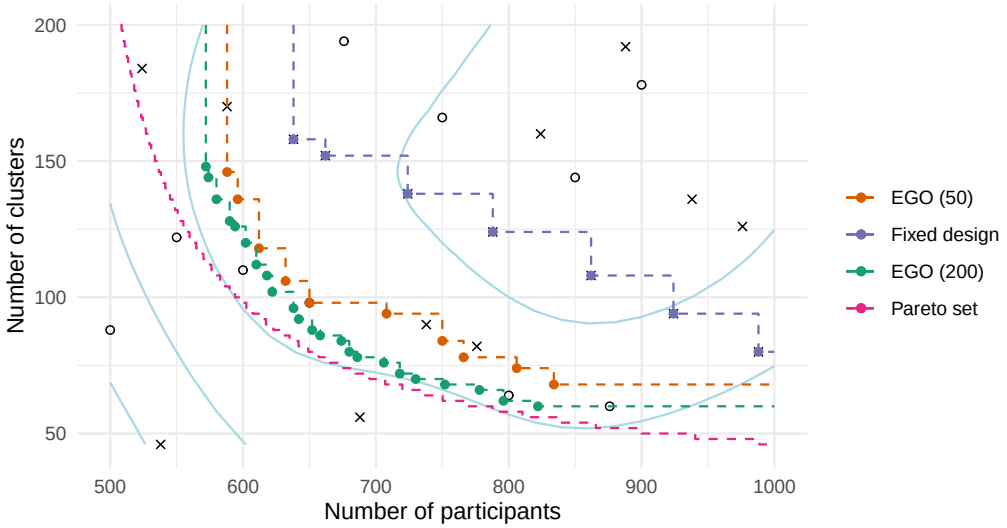


**Figure 4.** Objective values of solutions in the initial set $\mathcal{X}_E$ (open circles), subsequent iterations of the algorithm (crosses), and those in the final approximations of the Pareto set (filled circles). The approximation sets are obtained using the EGO algorithm for 50 and 200 evaluations, and using a fixed design of size 50). Contours represent the mean function of the final GP model.

To apply the proposed method, we define $k$ and $n = m * k$ as design parameters and search over the ranges $k \in [10, 100]$ and $n \in [100, 500]$. We wish to minimise the two objectives $f_1(\mathbf{x}) = 2n$ (the total number of participants) and $f_2(\mathbf{x}) = 2k$ (the total number of clusters). Fixing the type I error rate at 0.025 (one-sided), we can then use simulation to estimate the type II error rate and find an approximation set of solutions which give a value no more than the nominal 0.1. We initialised the optimisation algorithm by generating a Sobol sequence of size 20 and computing MC estimates of power for each point using $N = 100$ samples. Following this, 30 iterations of the algorithm were applied, with $N = 100$ samples used at each iteration. To explore the effect of increasing the computational budget, we then continued with another 150 iterations of the algorithm. For comparison, we found an alternative approximation set using the approach of Section 4.4, evaluating 50 solutions in a fixed space-filling design. Finally, we note that in this example power can be calculated analytically (by noting that the variance of the cluster means will equal $\sigma_B^2 + \sigma_W^2/m$), and so we can find the true Pareto set and compare the three approximation sets against it.

In Figure 4 we plot the 50 solutions evaluated over the course of the EGO algorithm, distinguishing between those in the initial design $\mathcal{X}_E$, those which were subsequently evaluated during the iterative phase of the algorithm, and those which together form the final approximation set. The contours of the mean function of the final GP model are also shown. We also plot the approximation front obtained using the comparator approach, the true Pareto set, and the approximation set obtained by running a further 150 iterations of the EGO algorithm.

**Table 1.** Approximation sets for the hypothetical example obtained using the EGO algorithm and the fixed design approach, with 50 evaluations each. Solutions are defined by the total number of participants, $2n$, and the total number of clusters, $2k$. The true type II error rate $\beta$ is shown alongside with the estimate $\hat{\beta}$ obtained using $N = 100$ MC samples.

| EGO | | | | Fixed design | | | |
|---|---|---|---|---|---|---|---|
| $2n$ | $2k$ | $\hat{\beta}$ (s.e.) | $\beta$ | $2n$ | $2k$ | $\hat{\beta}$ (s.e.) | $\beta$ |
| 834 | 68 | 0.06 (0.026) | 0.073 | 988 | 80 | 0.04 (0.019) | 0.039 |
| 806 | 74 | 0.07 (0.025) | 0.070 | 924 | 94 | 0.05 (0.019) | 0.036 |
| 766 | 78 | 0.11 (0.026) | 0.073 | 862 | 108 | 0.05 (0.019) | 0.036 |
| 750 | 84 | 0.10 (0.026) | 0.070 | 788 | 124 | 0.03 (0.02) | 0.040 |
| 708 | 94 | 0.06 (0.026) | 0.071 | 724 | 138 | 0.02 (0.021) | 0.047 |
| 650 | 98 | 0.13 (0.028) | 0.083 | 662 | 152 | 0.05 (0.023) | 0.056 |
| 632 | 106 | 0.07 (0.028) | 0.083 | 638 | 158 | 0.05 (0.024) | 0.061 |
| 612 | 118 | 0.07 (0.028) | 0.083 | | | | |
| 596 | 136 | 0.07 (0.027) | 0.080 | | | | |
| 588 | 146 | 0.08 (0.027) | 0.080 | | | | |

We find that the EGO algorithm leads to a much better approximation set than the comparator, for the same computation budget. For example, the fixed design method suggests that in order to limit the number of clusters to 80, a total of 988 participants will be required. In contrast, the EGO method shows that a total sample size of 766. The true minimal sample size required for 80 clusters is 649 participants, showing that while the EGO algorithm can substantially improve upon the simpler alternative, it has still suggested an approximation set that is some distance from the Pareto set. As shown in Table 1, all the suggested solutions are adequately powered with a true type II error rate $\beta$ less than the nominal 0.1. Increasing the computational budget by applying a further 150 iterations of the algorithm moves the approximation set closer to the Pareto set, as would be expected.

## *5.2 Simulation study*

To understand the variability in performance of the proposed method, we conducted a simulation study based on the preceding hypothetical example in Section 5.1. We applied the method as before, with 20 initial evaluations at points determined by a Sobol sequence, followed by 30 iterations of the algorithm. At each evaluation, $N = 100$ MC samples were used when estimating power. We repeated this process 500 times. For comparison, we also applied the fixed design approach by evaluating 50 solutions chosen by a Sobol sequence, discarding those with an estimated type II error rate greater than the nominal 0.1, and finding the approximation set from the remaining solutions. We repeated this process 500 times, again using $N = 100$ MC samples at each evaluation. We then extended the fixed design from size 50 to size 200, 400, 600, 800 and 1000. For each approximation set obtained, we recorded the dominated hypervolume and the number of solutions it contains. We also calculated the true type II error rate for each solution, and recorded the proportion of solutions in the approximation set which had a true type II error rate no more than the nominal 0.1.

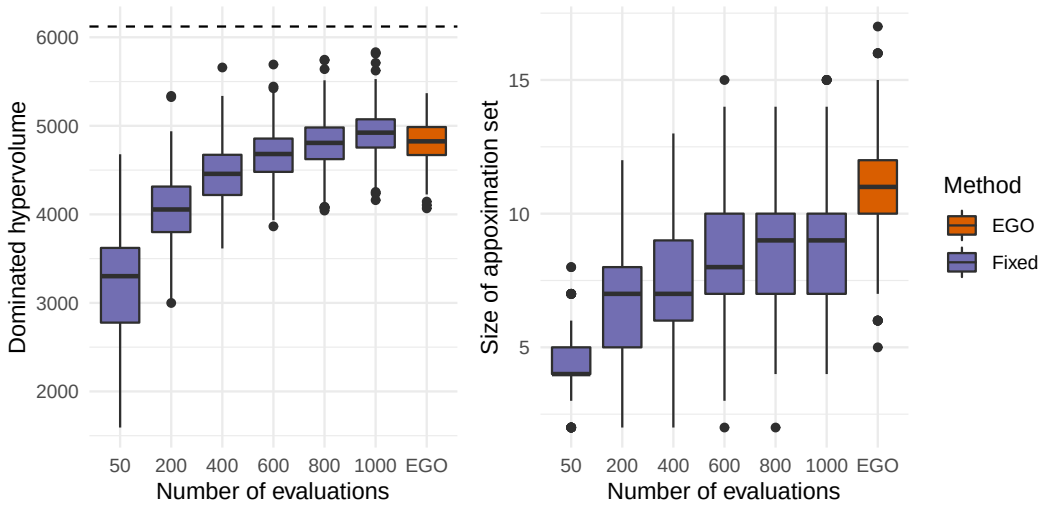


**Figure 5.** Distributions of dominated hypervolumes (left) and size (right) of approximation sets obtained using the EGO method and the fixed design method. The dashed horizontal line on the left hand plot indicates the dominated hypervolume of the true Pareto set.

We plot the distribution of the dominated hypervolumes in Figure 5. Comparing the EGO and fixed design methods when each use 50 evaluations in total, we find that the average dominated hypervolumes are 4824 and 3243 respectively. There is some overlap in the distributions. For example, 92% of EGO runs led to a dominated hypervolume greater than 4500, compared with 1% of fixed design runs. Increasing the number of evaluations in the fixed design improves the solution quality as expected, but to obtain an average performance similar to the EGO algorithm the number of evaluations must be increased to between 800 and 1000.

Figure 5 shows that the approximation sets are typically larger than those obtained using fixed designs, with an average of 10.9 solutions. In comparison, a fixed design of size 50 had an average of 4.4. This suggests that as well as producing approximation sets of consistently higher quality, the EGO approach will also provide more choice for trading off the two conflicting objectives. The vast majority (98%) of EGO approximation sets contained only valid solutions with a true type II error rate of at most 0.1. For the fixed design method, this proportion was similar when 50 solutions were evaluated but decreased as the number of evaluations increased. For fixed designs of size 1000, 63% of approximation sets contained at least one invalid solutions, although there were at most three invalid solutions in any one approximation set.

We also investigated the effect of changing $E$, the number of solutions evaluated initially before the iterative part of the EGO algorithm. Keeping the total number of evaluations at 50, we considered $E = 10$ and 30 in addition to the case $E = 20$ already presented. As before, the algorithm was run 500 times and the dominated hypervolume of the final approximation set was recorded. We found little difference between setting $E = 10$ (mean 4823) and $E = 20$ (mean

4824). When the number of initial evaluations was increased to $E = 30$, we found a statistically significant decrease in the mean dominated hypervolume to 4761. We can therefore conclude that, in this example, the recommendations referenced in Section 4.5 (setting $E$ to ten times the dimensions of the solutions space and representing 30 to 50% of the total computational budget) work well.

### 5.3 Application to the motivating example

Recall that the PACE trial measured two primary endpoints, one relating to fatigue (Denoted $F$) and one to disability (denoted $D$). Our model of the continuous response of the $i$th participant for endpoint $r \in \{F, D\}$ can be written, using the multilevel model notation of Goldstein [41], as

$$y_i^r = \beta_0^r + \beta_1^r t_i + u^r_{\text{therapist}(i)} t_i + v^r_{\text{doctor}(i)} + e_i^r \tag{19}$$

Correlation between the two endpoints is modelled by simulating bivariate residuals $(e_i^F, e_i^D)$ from a joint logistic distribution with correlation $\rho_W$ and marginal variances $\sigma_W^2$. We also allow for correlation between the random effects associated with each therapist and doctor. These are simulated according to the bivariate normal distributions

$$(u^F_{\text{therapist}(i)}, u^D_{\text{therapist}(i)}) \sim N\left((0,0)^T, \quad \begin{pmatrix} \sigma_T^2 & \rho_T\sigma_T^2 \\ \rho_T\sigma_T^2 & \sigma_T^2 \end{pmatrix} \quad \right)$$

$$(v^F_{\text{doctor}(i)}, v^D_{\text{doctor}(i)}) \sim N\left((0,0)^T, \quad \begin{pmatrix} \sigma_D^2 & \rho_D\sigma_D^2 \\ \rho_D\sigma_D^2 & \sigma_D^2 \end{pmatrix} \quad \right)$$

For the purposes of power calculations we must make some assumptions about the various nuisance parameter values. We set the 2nd level variance components to $\sigma_T^2 = 0.19, \sigma_D^2 = 0.37$ and $\sigma_W^2 = 3.29$ in order to give a variance partition coefficient of $\sigma_D^2/(\sigma_D^2 + \sigma_W^2) = 0.1$ in the control arm, a typical value in this setting. Similarly, the variance partition coefficient for between-therapist variance is then $\sigma_T^2/(\sigma_T^2 + \sigma_D^2 + \sigma_W^2) = 0.05$, and for between-doctor variation, $\sigma_D^2/(\sigma_T^2 + \sigma_D^2 + \sigma_W^2) = 0.095$. We set all correlations equal at $\rho_W = \rho_T = \rho_D = \rho_D = 0.9$, reflecting a situation where both a patient's responses and the individual therapist and doctor effects and very similar for both the fatigue and disability outcomes. We want to simulate the power of the trial under the alternative hypotheses $H_1 : \beta_1 = 1.10$. Note that this corresponds to a treatment effect standardised with respect to the total standard deviation in the intervention arm of $1.10/\sqrt{(0.19 + 0.37 + 3.29)} = 0.56$.

Our design parameters (together with the ranges considered) are the total sample size in the intervention arm, denoted $n_1$ (50 to 100); the number of therapists, $k$ (2 to 10); the allocation ratio relating the total number of participants in each arm, $r = n_0/n_1$ (0.5 to 1.5); the number of doctors, $j$ (3 to 20); and the nominal type I error rate to be used in the hypothesis tests, $a$ (0.05 to 0.2). The three objective functions to be minimised are $f_1(\mathbf{x}) = n_1 + rn_1$, $f_2(\mathbf{x}) = k$ and $f_3(\mathbf{x}) = j$. The two constraints to be satisfied are $g_1(\mathbf{x}) = \beta(\mathbf{x}) - 0.1$ and $g_2(\mathbf{x}) = \alpha(\mathbf{x}) - 0.2$.

We initialised the algorithm by generating a Sobol sequence of size 50 and computing MC estimates of power for each point using $N = 100$ MC samples. After 150 iterations of the algorithm, an approximation set containing 12 solutions was obtained. The algorithm took 2

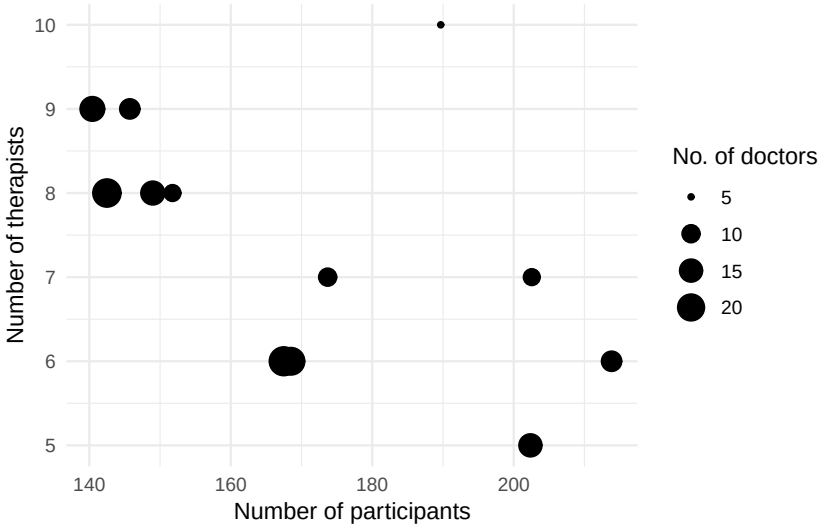


**Figure 6.** Objective values of the approximation set obtained following 50 iterations of the algorithm for the motivating example.

hours and 36 minutes to run on a PC with a 2.60GHz processor and 8GB RAM. The objective values of these solutions are illustrated in Figure 6, with full details provided in Table 2. The total number of participants ranged from 140 to 214; of therapists, from 5 to 10; and of doctors, from 5 to 23. Type I error rates ranged from 0.09 to 0.14, all some way below the actual constraint value of 0.2. We calculated precise MC estimates (using $N = 50^4$ samples) of both type I and II error rates for each solution in the approximation set. As shown in Table 2, type II error rates all appear to be around or slightly below the constraint of 0.1. This demonstrates the GP's ability to share information of MC estimates computed at several points to increase the precision at each of them. Type I error rates, in contrast, are in some cases significantly below the constraint of 0.2. This suggests there is some potential for improvement in the approximation set by applying further iterations of the algorithm. For comparison, we use the fixed design approach, evaluating 200 solutions, but found that all of these had a type I or II error rate beyond their respective constraints.

## 6 Discussion

Although simulation is often required for clinical trial sample size determination, related methodology has typically assumed that there is only one parameter which we are able to adjust (the sample size); that there is only one operating characteristic which must be estimated using simulation (the power of the trial); and that our goal is to minimise only one criterion (the sample size again)[3,5]. In this paper we have described a flexible approach to simulation-based SSD which can be used for more general multi-parameter problems. The method draws

**Table 2.** Approximation set after 50 iterations for the motivating example. Solutions are defined by the number of participants in the APT arm $n_1$, the total number of participants across both arms $n$, the number of therapists $k$, the number of doctors $j$, and the nominal type I error rates $a$. Both Type I and II error rates are estimated using simulation using $N$ samples, and are constrained at 0.2 and 0.1 respectively.

| | | | | | $N = 10^2$ | | $N = 50^4$ | |
|---|---|---|---|---|---|---|---|---|
| $n_1$ | $n$ | $k$ | $j$ | $a$ | $\beta$ (s.e.) | $\alpha$ (s.e.) | $\beta$ (s.e.) | $\alpha$ (s.e.) |
| 94 | 202 | 5 | 15 | 0.11 | 0.07 (0.026) | 0.12 (0.033) | 0.088 (0.003) | 0.17 (0.004) |
| 94 | 169 | 6 | 21 | 0.12 | 0.11 (0.031) | 0.13 (0.034) | 0.081 (0.003) | 0.179 (0.004) |
| 94 | 214 | 6 | 12 | 0.11 | 0.05 (0.022) | 0.21 (0.041) | 0.09 (0.003) | 0.157 (0.004) |
| 84 | 167 | 6 | 23 | 0.10 | 0.09 (0.029) | 0.12 (0.033) | 0.103 (0.003) | 0.147 (0.004) |
| 79 | 174 | 7 | 10 | 0.12 | 0.17 (0.038) | 0.13 (0.034) | 0.086 (0.003) | 0.171 (0.004) |
| 95 | 203 | 7 | 9 | 0.09 | 0.1 (0.03) | 0.16 (0.037) | 0.085 (0.003) | 0.134 (0.003) |
| 75 | 152 | 8 | 9 | 0.13 | 0.1 (0.03) | 0.12 (0.033) | 0.083 (0.003) | 0.175 (0.004) |
| 78 | 149 | 8 | 16 | 0.12 | 0.08 (0.027) | 0.21 (0.041) | 0.088 (0.003) | 0.171 (0.004) |
| 76 | 142 | 8 | 22 | 0.12 | 0.07 (0.026) | 0.17 (0.038) | 0.098 (0.003) | 0.156 (0.004) |
| 80 | 140 | 9 | 17 | 0.13 | 0.04 (0.02) | 0.2 (0.04) | 0.084 (0.003) | 0.176 (0.004) |
| 81 | 146 | 9 | 12 | 0.14 | 0.14 (0.035) | 0.18 (0.039) | 0.079 (0.003) | 0.181 (0.004) |
| 97 | 190 | 10 | 5 | 0.14 | 0.07 (0.026) | 0.27 (0.045) | 0.072 (0.003) | 0.178 (0.004) |

on established global optimisation algorithms which use statistical 'surrogate' models to solve design problems where there are several parameters to be chosen, several objectives to minimise, and several constraints to satisfy. We have illustrated how such problems arise in clinical trials of complex interventions.

The general optimisation framework we have suggested recognises that in many complex trials we are interested in minimising more than one quantity subject to constraints on operating characteristics. Problems of this sort are common in multilevel trial design [42], but are typically approached by first reducing the multiple objectives down to a single objective. For example, in the design of a cluster randomised trial it is common to fix the number of participants per cluster and minimise the number of clusters [43], or vice-versa [44,45]. Alternatively, a function which specifies the cost of sampling at the cluster and the patient level could be specified [46], and the overall cost minimised [47]. The latter approach has been suggested for both two-level [48] and three-level hierarchical trial designs [49,50]. However, the *a priori* specification of such a cost function may not always be feasible, particularly when several stakeholders are involved [51]. The Pareto optimisation framework we have described leads to a more computationally challenging optimisation problem, but produces a set of good solutions enabling the available trade-offs between objectives to be seen and selected from.

As noted in Section 1, related work in simulation-based design methodology has often focussed on a specific area of application. One advantage that brings is the relative ease with which the software can be used to solve a new problem within the same area. In contrast, our approach requires that the user provides a program which simulates the data generation and analysis of their proposed trial design. Although some have argued that this requirement may be prohibitive in practice [18], it allows the user to solve their specific problem rather than some related version of

it. Moreover, prior to addressing the sample size issue, modelling and simulation can help inform many other aspects of trial design, such as the patient population or the choice of endpoint [52]. One way to assist users in writing their own simulations is to share example programs for a range of problems, providing a starting point for the development of a new program. We have provided some examples in the appendix.

When submitting a proposed design for approval by a funding body it is important that the sample size calculation is transparent and replicable. This may be achieved in the context of simulation-based SSD by supplying the simulation program as part of the application [5]. Given this, any reviewer should be able to re-calculate the operating characteristics of the proposed design. However, a greater challenge for the reviewer is understanding the program and ensuring it is an accurate representation of the model in question. This requirement has partly motivated our use of R. Although significantly slower than a compiled language such as C++, it has been argued that software written in R is more transparent [52]. Validation will be further facilitated if a simulation protocol of the sort described in [53] is provided alongside the code. Future work could develop an interface for alternative statistical software such as Stata or SAS, allowing a simulation program to be written in them and connect with the R implementation of the optimisation algorithm.

We have followed the conventional approach to clinical trial design whereby constraints on operating characteristics are set and then a constrained optimisation problem is solved. In practice the constraints are not fixed in advance, but adjusted iteratively in response to the design requirements they produce. For example, an initial nominal power of 90% may require an infeasibly large sample size, leading to a revision down to 80%. Such an iterative procedure will increase an already substantial computational burden for simulation-based design. However, note that a change to a constraint does not mean starting the process again, since any previous MC estimates can still be used when fitting the GP model(s). The sequential nature of the optimisation algorithm suggests that an interactive routine could be developed, where the user pauses the algorithm in response to the sample size requirements which are being observed, adjusts the constraints, and then continues with the optimisation.

We have defined power in relation to a specific value of the parameter of interest, the so-called Minimal Clinically Important Difference, and specific point estimates of any nuisance parameters. Implicit in this formulation is an assumed monotonicity of the power function with respect to the parameter of interest. That is, we assume that if the power at this point is above the nominal constraint, then power at all larger values will also be above this level. When this assumption cannot be made, the method could be applied by setting further power constraints at other points in the parameter space. This approach could also be taken with respect to any nuisance parameters, helping to guard against any error in their estimation. This will, however, increase the computational burden by requiring more simulations at each iteration of the algorithm.

The examples have demonstrated that the time required to solve a sample size determination problem can be significant, of the order of hours. Given that the majority of computational effort is expended generating MC samples when evaluating solutions, it is important that these simulation programs are as efficient as possible. We recommend making use of code profilers such as R's 'Rprof' to identify the parts of the program that are consuming the most resources. Further efficiencies could potentially be gained by using more sophisticated methods for surrogate

modelling and efficient optimisation. For example, prior knowledge such as the power function being bounded or monotonic could be incorporated into the surrogate modelling process [29].

Numerous extensions to the proposed approach can be considered. One argument for simulation-based design is the ease with which sensitivity to model assumptions, such as the value of nuisance parameters, can be assessed [3]. Future work could consider how a systematic assessment of sensitivity to nuisance parameters could be conducted, given a proposed trial design. Such investigations fall under the heading of *uncertainty quantification* and can be carried out using GP regression and associated techniques [54]. A further extension could consider Bayesian approaches to trial design, including hybrid Bayesian-frequentist assurances [55], fully Bayesian measures such as average coverage criterion [56], and decision-theoretic methods [57]. Aside from very simple cases involving only conjugate analyses, evaluating these Bayesian criteria will generally require simulation [55] and so optimal design may benefit from the efficient methods discussed here. Complex SSD problems are also common in the area of adaptive designs, which can aim to minimise the expected sample size under several different hypotheses and over a number of stopping rule parameters [19]. Extending the proposed methods to such problems would require using surrogate models to approximate the objective functions, as opposed to only the constraints.

In conclusion, efficient optimisation algorithms based on surrogate models of expensive operating characteristic functions can be used to solve complex clinical trial sample size determination problems. By using these methods we can avoid making unrealistic simplifying assumptions at the trial design stage, both in terms of the statistical model underlying the trial and of the nature of the design problem.

## Declaration of conflicting interests

The Authors declare that there is no conflict of interest.

## Funding

This work was supported by the Medical Research Council [grant number MR/N015444/1].

## References

1. Cook JA, Julious SA, Sones W et al. DELTA2 guidance on choosing the target difference and undertaking and reporting the sample size calculation for a randomised controlled trial. *Trials* 2018; 19(1). DOI:10.1186/s13063-018-2884-0.
2. Arnold BF, Hogan DR, Colford JM et al. Simulation methods to estimate design power: an overview for applied research. *BMC Med Res Methodol* 2011; 11(1). DOI:10.1186/1471-2288-11-94. URL `http://dx.doi.org/10.1186/1471-2288-11-94`.
3. Landau S and Stahl D. Sample size and power calculations for medical studies by simulation when closed form expressions are not available. *Statistical Methods in Medical Research* 2013; 22(3): 324–345. DOI:10.1177/0962280212439578. URL `http://smm.sagepub.com/content/22/3/324.abstract`. `http://smm.sagepub.com/content/22/3/324.full.pdf+html`.

4. Feng Z and Grizzle JE. Correlated binomial variates: Properties of estimator of intraclass correlation and its effect on sample size calculation. *Statistics in Medicine* 1992; 11(12): 1607–1614. DOI: 10.1002/sim.4780111208. URL `http://dx.doi.org/10.1002/sim.4780111208`.
5. Hooper R. Versatile sample-size calculation using simulation. *The STATA Journal* 2013; 13(1): 21–38. URL `http://www.stata-journal.com/article.html?article=st0282`.
6. Schoenfeld DA and Borenstein M. Calculating the power or sample size for the logistic and proportional hazards models. *Journal of Statistical Computation and Simulation* 2005; 75(10): 771–785. DOI:10.1080/00949650410001729445. URL `http://dx.doi.org/10.1080/00949650410001729445`.
7. Grieve AP and Sarker SJ. Simulation-based sample-sizing and power calculations in logistic regression with partial prior information. *Pharmaceutical Statistics* 2016; 15(6): 507–516. DOI: 10.1002/pst.1773. URL `http://dx.doi.org/10.1002/pst.1773`.
8. Sutton AJ, Cooper NJ, Jones DR et al. Evidence-based sample size calculations based upon updated meta-analysis. *Statistics in Medicine* 2007; 26(12): 2479–2500. DOI:10.1002/sim.2704. URL `https://doi.org/10.1002%2Fsim.2704`.
9. Fedorov V and Jones B. The design of multicentre trials. *Statistical Methods in Medical Research* 2005; 14(3): 205–248. DOI:10.1191/0962280205sm399oa. URL `https://doi.org/10.1191/0962280205sm399oa`. PMID: 15969302, `https://doi.org/10.1191/0962280205sm399oa`.
10. Baio G, Copas A, Ambler G et al. Sample size calculation for a stepped wedge trial. *Trials* 2015; 16(1). DOI:10.1186/s13063-015-0840-9. URL `http://dx.doi.org/10.1186/s13063-015-0840-9`.
11. Hooper R, Teerenstra S, de Hoop E et al. Sample size calculation for stepped wedge and other longitudinal cluster randomised trials. *Statistics in Medicine* 2016; 35(26): 4718–4728. DOI: 10.1002/sim.7028. URL `http://dx.doi.org/10.1002/sim.7028`.
12. Reich NG, Myers JA, Obeng D et al. Empirical power and sample size calculations for cluster-randomized and cluster-randomized crossover studies. *PLOS ONE* 2012; 7(4): 1–7. DOI: 10.1371/journal.pone.0035564. URL `https://doi.org/10.1371/journal.pone.0035564`.
13. Wilson DT, Walwyn RE, Brown J et al. Statistical challenges in assessing potential efficacy of complex interventions in pilot or feasibility studies. *Statistical Methods in Medical Research* 2015; 25(3): 997–1009. DOI:10.1177/0962280215589507. URL `http://dx.doi.org/10.1177/0962280215589507`.
14. Feiveson AH. Power by simulation. *The Stata Journal* 2002; 2: 107–124. URL `http://ageconsearch.umn.edu/bitstream/115955/2/sjart_st0010.pdf`.
15. Browne WJ, Lahi MG and Parker RM. *A Guide to Sample Size Calculations for Random Effect Models via Simulation and the MLPowSim Software Package*, 2009. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.408.6314&rep=rep1&type=pdf`.
16. Williams MS, Ebel ED and Wagner BA. Monte carlo approaches for determining power and sample size in low-prevalence applications. *Preventive Veterinary Medicine* 2007; 82(1-2): 151–158. DOI: 10.1016/j.prevetmed.2007.05.015.
17. Jung SH. Sample size calculation for paired survival data: a simulation method. *Journal of Statistical Computation and Simulation* 2008; 78(1): 85–92. DOI:10.1080/10629360600864951.
18. Kontopantelis E, Springate DA, Parisi R et al. Simulation-based power calculations for mixed effects modeling: ipdpower in stata. *Journal of Statistical Software* 2016; 74(12). DOI:10.18637/jss.v074.i12. URL `http://dx.doi.org/10.18637/jss.v074.i12`.

19. Wason JMS and Jaki T. Optimal design of multi-arm multi-stage trials. *Statistics in Medicine* 2012; 31(30): 4269–4279. DOI:10.1002/sim.5513. URL http://dx.doi.org/10.1002/sim.5513.
20. Deb K, Pratap A, Agarwal S et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 2002; 6(2): 182–197. DOI:10.1109/4235.996017. URL http://dx.doi.org/10.1109/4235.996017.
21. Mersmann O. *mco: Multiple Criteria Optimization Algorithms and Related Functions*, 2014. URL https://CRAN.R-project.org/package=mco. R package version 1.0-15.1.
22. Sacks J, Welch WJ, Mitchell TJ et al. Design and analysis of computer experiments. *Statistical Science* 1989; 4(4): 409–423. URL http://www.jstor.org/stable/2245858.
23. Santner TJ, Williams BJ and Notz WI. *The Design and Analysis of Computer Experiments*. Springer-Verlag New York, Inc., 2003.
24. Krige D. A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy* 1951; 52(6): 119–139. URL https://journals.co.za/content/saimm/52/6/AJA0038223X_4792.
25. Rasmussen CE and Williams CKI. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
26. Roustant O, Ginsbourger D and Deville Y. DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of Statistical Software* 2012; 51(1): 1–55. URL http://www.jstatsoft.org/v51/i01/.
27. Jones DR. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization* 2001; 21(4): 345–383. DOI:10.1023/A:1012771025575. URL http://link.springer.com/article/10.1023/A:1012771025575.
28. Sasena MJ, Papalambros P and Goovaerts P. Exploration of metamodeling sampling criteria for constrained global optimization. *Engineering Optimization* 2002; 34(3): 263–278. DOI: 10.1080/03052150211751. URL https://doi.org/10.1080%2F03052150211751.
29. Emmerich MT, Deutz AH and Klinkenberg JW. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*. IEEE, pp. 2147–2154. DOI:10.1109/cec.2011.5949880.
30. Picheny V and Ginsbourger D. Noisy kriging-based optimization methods: A unified implementation within the DiceOptim package. *Computational Statistics & Data Analysis* 2014; 71: 1035–1053. DOI: 10.1016/j.csda.2013.03.018. URL https://doi.org/10.1016%2Fj.csda.2013.03.018.
31. White P, Sharpe M, Chalder T et al. Protocol for the pace trial: A randomised controlled trial of adaptive pacing, cognitive behaviour therapy, and graded exercise as supplements to standardised specialist medical care versus standardised specialist medical care alone for patients with the chronic fatigue syndrome/myalgic encephalomyelitis or encephalopathy. *BMC Neurology* 2007; 7(1): 6. DOI: 10.1186/1471-2377-7-6. URL http://www.biomedcentral.com/1471-2377/7/6.
32. White P, Goldsmith K, Johnson A et al. Comparison of adaptive pacing therapy, cognitive behaviour therapy, graded exercise therapy, and specialist medical care for chronic fatigue syndrome (pace): a randomised trial. *The Lancet* 2011; 377(9768): 823–836. DOI:10.1016/S0140-6736(11)60096-2. URL http://www.thelancet.com/journals/lancet/article/PIIS0140-6736(11)60096-2/abstract.
33. Walwyn REA and Roberts C. Therapist variation within randomised trials of psychotherapy: implications for precision, internal and external validity. *Statistical Methods in Medical Research* 2010; 19(3): 291–315. DOI:10.1177/0962280209105017. URL http://smm.sagepub.com/content/19/3/291.abstract. http://smm.sagepub.com/content/19/3/291.full.pdf+html.

34. Senn S and Bretz F. Power and sample size when multiple endpoints are considered. *Pharmaceut Statist* 2007; 6(3): 161–170. DOI:10.1002/pst.301. URL `http://dx.doi.org/10.1002/pst.301`.
35. Jones DR, Schonlau M and Welch WJ. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 1998; 13(4): 455–492. DOI:10.1023/a:1008306431147. URL `http://dx.doi.org/10.1023/A:1008306431147`.
36. Bendtsen C. *pso: Particle Swarm Optimization*, 2012. URL `https://CRAN.R-project.org/package=pso`. R package version 1.0.3.
37. Picheny V, Ginsbourger D and Richet Y. Noisy expected improvement and on-line computation time allocation for the optimization of simulators with tunable fidelity. In *2nd International Conference on Engineering Optimization*. URL `https://hal.archives-ouvertes.fr/hal-00489321v2/document`.
38. Dutang C and Savicky P. *randtoolbox: Generating and Testing Random Numbers*, 2015. R package version 1.17.
39. Chevalier C, Picheny V and Ginsbourger D. KrigInv: An efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging. *Computational Statistics & Data Analysis* 2014; 71: 1021–1034. DOI:10.1016/j.csda.2013.03.008. URL `http://dx.doi.org/10.1016/j.csda.2013.03.008`.
40. Bates D, Mächler M, Bolker B et al. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software* 2015; 67(1): 1–48. DOI:10.18637/jss.v067.i01.
41. Goldstein H. *Multilevel Statistical Models*. 3rd ed. Arnold, 2003.
42. Hemming K, Eldridge S, Forbes G et al. How to design efficient cluster randomised trials. *BMJ* 2017; 358. DOI:10.1136/bmj.j3064. URL `http://www.bmj.com/content/358/bmj.j3064`. `http://www.bmj.com/content/358/bmj.j3064.full.pdf`.
43. Donner A and Klar N. *Design and Analysis of Cluster Randomization Trials in Health Research*. London Arnold Publishers, 2000.
44. Hemming K, Girling A, Sitch A et al. Sample size calculations for cluster randomised controlled trials with a fixed number of clusters. *BMC Medical Research Methodology* 2011; 11(1): 102. DOI:10.1186/1471-2288-11-102. URL `http://www.biomedcentral.com/1471-2288/11/102`.
45. Eldridge SM, Costelloe CE, Kahan BC et al. How big should the pilot study for my cluster randomised trial be? *Statistical Methods in Medical Research* 2015; DOI:10.1177/0962280215588242. URL `http://smm.sagepub.com/content/early/2015/06/12/0962280215588242.abstract`. `http://smm.sagepub.com/content/early/2015/06/12/0962280215588242.full.pdf+html`.
46. Hox J. *Multilevel Analysis: Techniques and Applications*. Lawrence Erlbaum Associates, Inc., 2002.
47. Snijders TAB and Bosker RJ. Standard errors and sample sizes for two-level research. *Journal of Educational and Behavioral Statistics* 1993; 18(3): 237–259. DOI:10.3102/10769986018003237. URL `http://jeb.sagepub.com/content/18/3/237.abstract`. `http://jeb.sagepub.com/content/18/3/237.full.pdf+html`.
48. Raudenbush SW and Liu X. Statistical power and optimal design for multisite randomized trials. *Psychological Methods* 2000; 5(2): 199–213. DOI:10.1037/1082-989x.5.2.199. URL `http://dx.doi.org/10.1037/1082-989X.5.2.199`.
49. van Breukelen GJ and Candel MJ. Calculating sample sizes for cluster randomized trials: We can keep it simple and efficient! *Journal of Clinical Epidemiology* 2012; 65(11): 1212 – 1218. DOI:http://dx.doi.org/10.1016/j.jclinepi.2012.06.002. URL `http://www.sciencedirect.com/science/article/pii/S0895435612001692`.

50. Teerenstra S, Moerbeek M, van Achterberg T et al. Sample size calculations for 3-level cluster randomized trials. *Clinical Trials* 2008; 5(5): 486–495. DOI:10.1177/1740774508096476. URL http://dx.doi.org/10.1177/1740774508096476.
51. Joseph L and Wolfson DB. Interval-based versus decision theoretic criteria for the choice of sample size. *Journal of the Royal Statistical Society: Series D (The Statistician)* 1997; 46(2): 145–149. DOI:10.1111/1467-9884.00070. URL http://dx.doi.org/10.1111/1467-9884.00070.
52. Smith MK and Marshall A. Importance of protocols for simulation studies in clinical drug development. *Statistical Methods in Medical Research* 2010; DOI:10.1177/0962280210378949. URL http://smm.sagepub.com/content/early/2010/08/03/0962280210378949.abstract. http://smm.sagepub.com/content/early/2010/08/03/0962280210378949.full.pdf+html.
53. Burton A, Altman DG, Royston P et al. The design of simulation studies in medical statistics. *Statistics in Medicine* 2006; 25(24): 4279–4292. DOI:10.1002/sim.2673. URL http://dx.doi.org/10.1002/sim.2673.
54. Kennedy MC and O'Hagan A. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 2001; 63(3): 425–464. DOI:10.1111/1467-9868.00294. URL http://dx.doi.org/10.1111/1467-9868.00294.
55. O'Hagan A, Stevens JW and Campbell MJ. Assurance in clinical trial design. *Pharmaceutical Statistics* 2005; 4(3): 187–201. DOI:10.1002/pst.175. URL http://dx.doi.org/10.1002/pst.175.
56. Cao J, Lee JJ and Alber S. Comparison of bayesian sample size criteria: ACC, ALC, and WOC. *Journal of Statistical Planning and Inference* 2009; 139(12): 4111 – 4122. DOI:http://dx.doi.org/10.1016/j.jspi.2009.05.041. URL http://www.sciencedirect.com/science/article/pii/S0378375809001670.
57. Oakley JE, Brennan A, Tappenden P et al. Simulation sample sizes for monte carlo partial evpi calculations. *Journal of Health Economics* 2010; 29(3): 468 – 477. DOI:https://doi.org/10.1016/j.jhealeco.2010.03.006. URL http://www.sciencedirect.com/science/article/pii/S0167629610000470.