Tempered SMC for Sequential Bayesian Optimal Design

Jacopo IOLLO[1,2,3] Christophe HEINKELE[3] Pierre ALLIEZ[2] Florence FORBES[1]

[1,2]Inria center at Université Grenoble Alpes and Inria center at Université Côte d'Azur, France

[3]Cerema, Strasbourg, France

Résumé – Nous proposons une approche Monte Carlo séquentielle tempérée (SMC) pour l'optimisation séquentielle du design experimental dans un context bayésien. Le choix séquentiel des parametres de design est effectué par une procédure d'approximation stochastique (SA) incorporant des échantillonneurs SMC avec tempering. Le tempering rend possible à la fois un gain d'information important et un échantillonnage SMC précis. Cette combinaison originale de SA et SMC permet d'obtenir simultanément le design optimal et une approximation de la loi a posteriori des paramètres. Une application à une tache de localisation de sources illustre qu'un gain intéressant peut être atteint en utilisant des simulations pour limiter le nombre de mesures couteuses à faire sur le terrain.

Abstract – We propose a tempered Sequential Monte Carlo (SMC) approach to sequential Bayesian optimal experimental design. The sequential design process is carried out through a Stochastic Approximation (SA) procedure using tempered SMC samplers. The tempering makes possible both a large information gain and an accurate SMC sampling. This novel combination of SA and SMC allows to simultaneously address the design optimization and the parameter posterior distribution approximation. An illustration, on a source localisation task, shows the approach potential using off line computer simulations to significantly reduce the number of costly field measurements.

1 Sequential Bayesian Optimal Experimental Design (BOED)

A design refers to some experimental conditions required to perform an experiment and get observations from the phenomenon under study. Experimental design can address allocating resources for information gathering, improving precision and/or prediction or reducing experimental costs. In this work, we assume that the design is determined by some parameter $\boldsymbol{\xi} \in \mathbb{R}^d$, representing for instance a location or a frequency at which we wish to measure a quantity. The desired designs are those that maximize information on some parameters of interest $\boldsymbol{\theta} \in \mathbb{R}^m$. In this context, the Bayesian framework is a unified way to account for prior information via a probability distribution $p(\boldsymbol{\theta})$, for uncertainties about the observations $\mathbf{y}$ through a distribution $p(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\xi})$, and for a design criterion (also called utility function) $F(\boldsymbol{\xi}, \boldsymbol{\theta}, \mathbf{y})$ describing the experimental aims. The prior is assumed to be independent of $\boldsymbol{\xi}$ and $p(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\xi})$ available in closed-form. However, such a Bayesian modelling often leads to an intractable joint optimization and integration. We propose to handle this issue by coupling a Stochastic Approximation (SA) with an efficient Sequential Monte Carlo approach (SMC).

1.1 Expected Information Gain (EIG)

There exist various utility functions F depending on the targeted task [15]. In this work, we focus on parameter estimation and consider an information-based utility leading to the so-called Expected Information Gain (EIG). The EIG, denoted by I, admits several equivalent expressions (see *e.g.* [10]). For instance, it can be written as the expected loss in entropy when accounting for an observation $\mathbf{y}$ at $\boldsymbol{\xi}$ or as a mutual information, using $p(\mathbf{y}, \boldsymbol{\theta}|\boldsymbol{\xi}) = p(\boldsymbol{\theta}|\mathbf{y}, \boldsymbol{\xi})p(\mathbf{y}|\boldsymbol{\xi}) = p(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\xi})p(\boldsymbol{\theta})$,

$$I(\boldsymbol{\xi}) = \mathbb{E}_{p(\mathbf{y}|\boldsymbol{\xi})}[H(p(\boldsymbol{\theta})) - H(p(\boldsymbol{\theta}|\mathbf{Y}, \boldsymbol{\xi})] \tag{1}$$

$$= \mathbb{E}_{p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta},\boldsymbol{\xi})}\left[\log \frac{p(\boldsymbol{\theta}|\mathbf{Y}, \boldsymbol{\xi})}{p(\boldsymbol{\theta})}\right] \tag{2}$$

where random variables are indicated with uppercase letters, $\mathbb{E}_p[\cdot]$ denotes the expectation with respect to p and $H(p(\boldsymbol{\theta})) = -\mathbb{E}_{p(\boldsymbol{\theta})}[\log p(\boldsymbol{\theta})]$ is the entropy of p. We thus look for $\boldsymbol{\xi}^*$ satisfying

$$\boldsymbol{\xi}^* = \arg\max_{\boldsymbol{\xi} \in \mathbb{R}^d} I(\boldsymbol{\xi}) \ . \tag{3}$$

Before optimizing $I(\boldsymbol{\xi})$, in most cases, evaluating $I(\boldsymbol{\xi})$ is difficult due to the intractability of $p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi})$ and $p(\boldsymbol{y}|\boldsymbol{\xi})$ [3].

1.2 Sequential design

Solving (3) is a *static* or *one-step* design problem. A single $\boldsymbol{\xi}$ or multiple $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_K\}$ are selected prior to any observation, measurements $\{\mathbf{y}_1, \dots, \mathbf{y}_K\}$ are made for these design parameters and the experiment is stopped. The prior $p(\boldsymbol{\theta})$ can be used to encode previous observations but in static design, the selected designs depend only on the model. In contrast, in sequential or iterated design, K experiments are planned sequentially to construct an adaptive strategy, meaning that for the kth experiment, the best $\boldsymbol{\xi}_k$ is selected taking into account the previous design parameters and associated observations $\boldsymbol{D}_{k-1} = \{(\mathbf{y}_1, \boldsymbol{\xi}_1), \dots (\mathbf{y}_{k-1}, \boldsymbol{\xi}_{k-1})\}$. Then, $\mathbf{y}_k$ is measured at $\boldsymbol{\xi}_k$ and $\boldsymbol{D}_k$ is updated into $\boldsymbol{D}_k = \boldsymbol{D}_{k-1} \cup (\mathbf{y}_k, \boldsymbol{\xi}_k)$. In practice, we adopt a greedy approach, choosing the next design $\boldsymbol{\xi}_k$ as if it was the last one, which consists of replacing in (1) the prior $p(\boldsymbol{\theta})$ by our current belief on $\boldsymbol{\theta}$, namely $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1}) = p(\boldsymbol{\theta}|\mathbf{y}_1, \boldsymbol{\xi}_1 \dots \mathbf{y}_{k-1}, \boldsymbol{\xi}_{k-1})$, and to solve iteratively for

$$\boldsymbol{\xi}_k^* = \arg\max_{\boldsymbol{\xi} \in \mathbb{R}^d} I_k(\boldsymbol{\xi}) \ , \tag{4}$$

$I_k(\boldsymbol{\xi}) = \mathbb{E}_{p(\mathbf{y}|\boldsymbol{\xi},\boldsymbol{D}_{k-1})}[H(p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})) - H(p(\boldsymbol{\theta}|\mathbf{Y},\boldsymbol{\xi},\boldsymbol{D}_{k-1}))]$. Observations are assumed conditionally independent, so that, $p(\boldsymbol{\theta}|\boldsymbol{D}_k) \propto p(\boldsymbol{\theta}) \prod_{i=1}^{k} p(\mathbf{y}_i|\boldsymbol{\theta},\boldsymbol{\xi}_i)$ which also leads to

$$p(\boldsymbol{\theta}|\boldsymbol{D}_k) \propto p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})\; p(\mathbf{y}_k|\boldsymbol{\theta},\boldsymbol{\xi}_k)\;. \tag{5}$$

1.3 EIG contrastive bound optimization

Going back to the optimization in (3), a standard gradient ascent algorithm would consist, at iteration t, of updating $\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t + \gamma_t\; \nabla_{\boldsymbol{\xi}} I(\boldsymbol{\xi})_{|\boldsymbol{\xi}=\boldsymbol{\xi}_t}$ with a stepsize γ_t, but in practice both $I(\boldsymbol{\xi})$ and its gradient $\nabla_{\xi} I(\boldsymbol{\xi})$ are intractable. However, they can both be expressed as expectations, which naturally leads to consider stochastic approximation approaches [2], among which the most popular is the stochastic gradient algorithm (SG). In a BOED setting, if $\nabla_{\boldsymbol{\xi}} I(\boldsymbol{\xi})$ is expressed as an expectation $\nabla_{\boldsymbol{\xi}} I(\boldsymbol{\xi}) = \mathbb{E}[f(\boldsymbol{\xi},\mathbf{X})]$ over a random variable $\mathbf{X}$, the SG iteration writes $\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t + \gamma_t\; f(\boldsymbol{\xi}_t, \mathbf{x}_t)$ with $\mathbf{x}_t$ a realisation of $\mathbf{X}$. This assumes that we can differentiate under the integral sign in (2). There exists different ways to differentiate, including the popular reparametrization trick, but SG for $I(\boldsymbol{\xi})$ remains difficult to perform due to the intractability of the integrand in (2). An alternative approach has been proposed in [10] referred to as a variational approximation. It consists of optimizing a tractable lower bound of $I(\boldsymbol{\xi})$ and computing $\boldsymbol{\xi}^*$ via an alternate maximization. In this work, we consider such a bound named the Prior Contrastive Estimation (PCE) bound and denoted by I_{PCE}. It is based on contrastive samples from L additional variables $\boldsymbol{\theta}_\ell$, for $\ell = 1 \dots L$, distributed following the prior $p(\boldsymbol{\theta})$, like $\boldsymbol{\theta}$ rewritten as $\boldsymbol{\theta}_0$. Denoting $F_{PCE}(\boldsymbol{\xi},\boldsymbol{\theta}_0,\cdot,\boldsymbol{\theta}_L,\mathbf{y}) = \log \frac{p(\mathbf{y}|\boldsymbol{\theta}_0,\boldsymbol{\xi})}{\frac{1}{L+1}\sum_{\ell=0}^{L} p(\mathbf{y}|\boldsymbol{\theta}_\ell,\boldsymbol{\xi})}$, I_{PCE} is defined as

$$I_{PCE}(\boldsymbol{\xi}) = \mathbb{E}_{p(y|\boldsymbol{\theta}_0,\boldsymbol{\xi})\; \bigotimes_{\ell=0}^{L} p(\boldsymbol{\theta}_\ell)}\left[F_{PCE}(\boldsymbol{\xi},\boldsymbol{\theta}_0,\cdot,\boldsymbol{\theta}_L,\mathbf{Y})\right]$$

The above quantity is a lower bound $I(\boldsymbol{\xi}) \geq I_{PCE}(\boldsymbol{\xi})$ and the bound is tight when L tends to ∞ (see [11] for a proof). It is tractable as all expressions $p(\mathbf{y}|\boldsymbol{\theta}_\ell,\boldsymbol{\xi})$ are tractable, and its gradient requires only the gradient $\nabla_{\boldsymbol{\xi}} p(\mathbf{y}|\boldsymbol{\theta},\boldsymbol{\xi})$. Stochastic approximation can be applied to maximize I_{PCE} as $\nabla_{\boldsymbol{\xi}} I_{PCE}(\boldsymbol{\xi})$ can be expressed as an expectation via a reparametrization trick. More specifically, we assume that there exists a transformation $T_{\boldsymbol{\xi},\boldsymbol{\theta}_0}$ such that $\mathbf{Y} = T_{\boldsymbol{\xi},\boldsymbol{\theta}_0}(U)$ with U independent of $\boldsymbol{\xi}$ and $\boldsymbol{\theta}_0$ and easy to simulate, *e.g.* U is a standard Gaussian. It follows under some mild conditions omitted here, that

$$\nabla_{\boldsymbol{\xi}} I_{PCE}(\boldsymbol{\xi}) = \mathbb{E}_{p(u)\; \bigotimes_{\ell=0}^{L} p(\boldsymbol{\theta}_\ell)}[\nabla_{\boldsymbol{\xi}} F_{PCE}(\boldsymbol{\xi},\boldsymbol{\theta}_0,\cdot,\boldsymbol{\theta}_L,T_{\boldsymbol{\xi},\boldsymbol{\theta}_0}(U))]$$

Only the differentiability of $F_{PCE}(\boldsymbol{\xi},\boldsymbol{\theta}_0,\cdot,\boldsymbol{\theta}_L,T_{\boldsymbol{\xi},\boldsymbol{\theta}_0}(U))$ in $\boldsymbol{\xi}$ within the expectation is required. A similar I_{PCE} bound and its gradient can be easily derived for greedy sequential design (4). Using the conditional independence assumption (5), at each step k, the prior $p(\boldsymbol{\theta})$ only needs to be replaced by the current posterior $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$. The stochastic gradient (SG) algorithm is described in Algorithm 1, with the additional possibility to estimate gradients with minibatches. In line 5, the current noisy gradient can actually be replaced by any unbiased estimate of the I_{PCE} gradient.

Optimizing the contrastive bound I_{PCE} requires then sampling $\boldsymbol{\theta}_0 \dots \boldsymbol{\theta}_L$ from $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$ (line 3). This is more costly than sampling from the prior as $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$ is only know up to a normalizing constant (5). Such simulations could be obtained via MCMC algorithms but the necessity to do so at each step would be too simulation intensive. As a more efficient alternative, we propose to use sequential Monte-Carlo (SMC) approaches for sequential sampling as detailed next.

Algorithm 1: SG with minibatches $(N_t)_{t=1:T}$ for (4)

1 Set T number of iterations, $\boldsymbol{\xi}_0$, stepsizes $(\gamma_t)_{t=1:T}$
2 **while** $t \leq T$ **do**
3 Sample $\boldsymbol{\theta}^i_{\ell,t} \sim p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$, for $\ell = 0:L, i = 1:N_t$
4 Sample $u^i_t \sim p(u)$, for $i = 1 : N_t$
5 Set $\nabla_{t+1} = \frac{1}{N_t}\sum_{i=1}^{N_t} \nabla_{\boldsymbol{\xi}} F_{PCE}(\boldsymbol{\xi}, \boldsymbol{\theta}^i_{0,t}, \cdot, \boldsymbol{\theta}^i_{L,t}, T_{\boldsymbol{\xi},\boldsymbol{\theta}^i_{0,t}}(u^i_t))|_{\boldsymbol{\xi}=\boldsymbol{\xi}_t}$
6 Update $\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t + \gamma_t \nabla_{t+1}$
7 **end**
8 **return** $\boldsymbol{\xi}^*_k = \boldsymbol{\xi}_T$ or a Polyak averaging value

2 Tempered Sequential Monte Carlo

SMC has been used in previous work on BOED mainly as an alternative to MCMC, to compute approximation of the EIG. For instance [7, 8] use SMC to incorporate model uncertainty or for finite-valued design, thus reducing optimization to a finite number of comparisons. In contrast, [14] propose SMC samplers to handle the optimisation part but they restrict to static design. In particular, their solution requires $p(\mathbf{y}|\boldsymbol{\xi})$ which is intractable and requires an approximation. This approximation is not easy to perform in a sequential setting. The originality of our approach is to consider continuous design parameters and to adopt a stochastic optimization approach to both compute and optimize the EIG. A tempered SMC is then used in conjunction with SA to efficiently sample the relevant quantities and estimate the noisy gradients required for the SG Algorithm 1 (lines 3 and 5).

More specifically, the goal is to provide samples from a sequence of probability distributions $\{p_k\}_{k=1:K}$. To simplify, we deal with probability densities assuming absolute continuity wrt the Lebesgue measure but the setting is more general, see [4]. An MCMC approach would require to build an ergodic kernel M_k and to run a Markov Chain for each p_k, which would be very compute intensive in the sequential context. In contrast, SMC samplers [6, 5, 4] provide the possibility to approximate p_{k+1} recycling samples from p_k. SMC samplers aim at propagating N samples also called particles $\boldsymbol{\theta}^{1:N}_k = \{\boldsymbol{\theta}^1_k, \dots \boldsymbol{\theta}^N_k\}$ and their corresponding weights $\mathbf{w}^{1:N}_k = \{w^1_k, \dots w^N_k\}$ in such a way that the empirical distribution p^N_k of the particles at times k converges to p_k: meaning that for all integrable function ϕ,

$$\mathbb{E}_{p^N_k}[\phi(\boldsymbol{\theta})] = \sum_{i=1}^{N} w^i_k \phi(\boldsymbol{\theta}^i_k) \xrightarrow[N\to\infty]{} \mathbb{E}_{p_k}[\phi(\boldsymbol{\theta})]$$

$$\text{with} \quad p^N_k(\cdot) = \sum_{i=1}^{N} w^i_k \delta_{\boldsymbol{\theta}^i_k}(\cdot) \quad \textit{particle approximation of } p_k.$$

As showed in [1], the number of particles N required to yield an accurate approximation p^N_k scales exponentially with the

Kullback-Leibler distance between the proposal p_{k-1} and the target p_k distributions. In a BOED context, this is potentially problematic as EIG-based design optimisation aims at increasing this distance by looking for design $\boldsymbol{\xi}_k$ that makes $p(\boldsymbol{\theta}|\boldsymbol{D}_k)$ as far as possible from the previous $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$, for higher information gain. Moving from $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$ to $p(\boldsymbol{\theta}|\boldsymbol{D}_k)$ with just one SMC step might then yield poor results. A solution is to consider *tempering* ([4], section 17.2.3) to move along a sequence of probability distributions interpolating between $p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$ and $p(\boldsymbol{\theta}|\boldsymbol{D}_k)$. A tempering path is a sequence of the form p_{λ_τ} with $0 = \lambda_0 < \lambda_\tau < \ldots < \lambda_\mathcal{T} = 1$ where $p_0 = p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})$ and $p_1 = p(\boldsymbol{\theta}|\boldsymbol{D}_k)$. Usually, only the initial and final distributions are imposed. Intermediate distributions p_{λ_τ} are not of interest so that the λ_τ's can be chosen as desired. A popular approach is to use what is know as the geometric path [5]: $p_{\lambda_\tau}(\boldsymbol{\theta}) \propto p_{k-1}(\boldsymbol{\theta})^{1-\lambda_\tau}\ p_k(\boldsymbol{\theta})^{\lambda_\tau}$, which using (5), in our context takes the form

$$p_{\lambda_\tau}(\boldsymbol{\theta}) \propto p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1})\ p(\mathbf{y}_k|\boldsymbol{\theta}, \boldsymbol{\xi}_k)^{\lambda_\tau}$$

or equivalently $p_{\lambda_\tau}(\boldsymbol{\theta}) = p_{\lambda_{\tau-1}}(\boldsymbol{\theta})\ p(\mathbf{y}_k|\boldsymbol{\theta}, \boldsymbol{\xi}_k)^{\lambda_\tau - \lambda_{\tau-1}}$.

As setting the sequence λ_τ manually can be a challenging task with disappointing results, we follow the adaptive strategy proposed by [12]. Given a user-set threshold ESS_{min} interpreted as an effective sample size, at iteration τ of the tempered SMC procedure, given a current set of particles $\boldsymbol{\theta}_\tau^{1:N}$, we set recursively $\lambda_\tau = \lambda_{\tau-1} + \delta$ with δ the solution in $[0, 1 - \lambda_{\tau-1}]$ of the following equation (if δ is not in $[0, 1 - \lambda_{\tau-1}]$, λ_τ is set to 1 and the tempering stops):

$$\frac{\left(\sum_{i=1}^N p\left(\mathbf{y}_k|\boldsymbol{\theta}_\tau^i, \boldsymbol{\xi}_k\right)^\delta\right)^2}{\sum_{i=1}^N p\left(\mathbf{y}_k|\boldsymbol{\theta}_\tau^i, \boldsymbol{\xi}_k\right)^{2\delta}} = ESS_{min}\ . \tag{6}$$

This is a relatively simple task to solve with numerical root finding. This procedure guarantees that the SMC approximation error remains stable over iterations, see [12] for details. It can be interpreted as a way to control the Chi-square pseudo-distance between the successive distributions, see [4] proposition 17.2. Tempered SMC then requires like SMC an unbiased resampling scheme denoted by $resample(\boldsymbol{\theta}^{1:N}, \mathbf{w}^{1:N})$. Resampling is the action of drawing randomly from a weighted sample, so as to obtain an unweighted sample. Several unbiased resampling schemes are listed in Chapter 9 of [4]. The most standard one is *multinomial* resampling which draws samples according to their weights, while *stratified* resampling has better variance properties. Tempered SMC also requires a family of MCMC kernels $(M_\lambda)_\lambda$ so that M_λ leaves p_λ invariant. The tempering, at step k of our sequential BOED, is specified in Algorithm 2 with a numerical illustration in Section 3.

Algorithm 2: Tempered SMC at step k

1 Set $\tau = 0, \lambda_0 = 0, N, ESS_{min}$, M_λ, and $resample$
2 Sample $\boldsymbol{\theta}_0^{1:N} \sim p(\boldsymbol{\theta}|\boldsymbol{D}_{k-1}) = p_{\lambda_0}(\boldsymbol{\theta})$
3 Set $w_0^i = 1/N$ for $i = 1:N$
4 **while** $\lambda_\tau < 1$ **do**
5 Set $\tau = \tau + 1$
6 Set $\tilde{\boldsymbol{\theta}}_{\tau-1}^{1:N} = resample(\boldsymbol{\theta}_{\tau-1}^{1:N}, \mathbf{w}_{\tau-1}^{1:N}))$ $(\sim p_{\lambda_{\tau-1}})$
7 Sample $\boldsymbol{\theta}_\tau^i \sim M_{\lambda_{\tau-1}}(\tilde{\boldsymbol{\theta}}_{\tau-1}^i, \cdot)$ for $i = 1:N$
8 Solve for δ, $\frac{\left(\sum_{i=1}^N p\left(\mathbf{y}_k|\boldsymbol{\theta}_\tau^i, \boldsymbol{\xi}_k\right)^\delta\right)^2}{\sum_{i=1}^N p\left(\mathbf{y}_k|\boldsymbol{\theta}_\tau^i, \boldsymbol{\xi}_k\right)^{2\delta}} = ESS_{min}$
9 Set $\lambda_\tau = \lambda_{\tau-1} + \delta$
10 Set $\tilde{w}_\tau^i = p(\mathbf{y}_k|\boldsymbol{\theta}_\tau^i, \boldsymbol{\xi}_k)^\delta$
11 and $w_\tau^i = \frac{\tilde{w}_\tau^i}{\sum_{j=1}^N \tilde{w}_\tau^j}$ for $i = 1:N$
12 **end**
13 **return** $\boldsymbol{\theta}_k^{1:N} = \boldsymbol{\theta}_\tau^{1:N}$ and $\mathbf{w}_k^{1:N} = \mathbf{w}_\tau^{1:N}$ for a particle approximation of $p(\boldsymbol{\theta}|\boldsymbol{D}_k) = p_1(\boldsymbol{\theta})$

3 Source localisation example

We consider the 2D location finding experiment described in [9]. It consists of S hidden sources in $\mathbb{R}^2$ whose locations $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_S\}$ are unknown. The number of source S is known. Each source emits a signal whose intensity attenuates according to the inverse-square law. The measured signal is the superposition of all these signals. The design problem is to choose where to make the measurements to best learn the source locations. If a measurement is performed at a point $\boldsymbol{\xi} \in \mathbb{R}^2$, the signal strength is $\mu(\boldsymbol{\theta}, \boldsymbol{\xi}) = b + \sum_{s=1}^S \frac{\alpha_s}{m + ||\boldsymbol{\theta}_s - \boldsymbol{\xi}||_2^2}$ where α_s, b and m are constants. A standard Gaussian prior is assumed for each $\boldsymbol{\theta}_s \sim \mathcal{N}(0, \boldsymbol{I})$ and the log total intensity is observed with some centered Gaussian noise with standard deviation σ. The likelihood is thus log-normal, *i.e.* $(\log \mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\xi}) \sim \mathcal{N}(\log \mu(\boldsymbol{\theta}, \boldsymbol{\xi}), \sigma)$. In this experiment, we set $S = 2$, $\alpha_1 = \alpha_2 = 1$, $m = 10^{-4}$, $b = 10^{-1}$, $\sigma = 0.5$ and we plan $K = 40$ successive design optimisations. The Markov kernel is that of a Metropolis-Hasting scheme with a Gaussian proposal centered at the current particle with a variance set to the empirical variance of the $\tilde{\boldsymbol{\theta}}_{\tau-1}^{1:N}$ (line 6). Resampling is done via a stratified scheme. We use $L = 20,000$ contrastive variables. At each step $k = 1:K$, we consider an I_{PCE}^k bound of $I^k(\boldsymbol{\xi})$. Algorithm 2 is used to get $N = 1,000$ simulations $\boldsymbol{\theta}_\ell^{1:N}$ of each contrastive variable. The Adam algorithm [13] is then used with standard hyperparameters to perform the stochastic gradient in Algorithm 1. Considering the large number of simulated $\boldsymbol{\theta}_\ell^{1:N}$, line 3 in Algorithm 1 is replaced by a random shuffling of these simulations. For comparison, we also consider the case where the observations $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ are simulated at random locations without design optimization.

The whole experiment is then repeated 150 times but drawing source locations at random each time. Figure 1 shows the cumulative EIG and the L_2 Wasserstein distances between weighted particles and the true source locations. Design optimization leads to a significant improvement both in terms of information gain and posterior estimation. Measurements can be reduced from 40 randomly located measurements to 13 with optimized locations, for the same approximation quality, as measured by the Wasserstein distance. Figure 2 illustrates the evolution of the particles over the design steps, starting from a sample following the prior to a sample concentrating around the true source locations. This provides a visual assessment of the quality of the posterior approximation.