Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design

Desi R. Ivanova*[1] Marcel Hedman*[1] Cong Guan[†] Tom Rainforth[1]
[1]*Department of Statistics, University of Oxford*

Abstract

We develop a semi-amortized, policy-based, approach to Bayesian experimental design (BED) called Step-wise Deep Adaptive Design (Step-DAD). Like existing, fully amortized, policy-based BED approaches, Step-DAD trains a design policy upfront before the experiment. However, rather than keeping this policy fixed, Step-DAD periodically updates it as data is gathered, refining it to the particular experimental instance. This allows it to improve both the adaptability and the robustness of the design strategy compared with existing approaches.

Keywords: Bayesian experimental design, adaptive design, information maximization

1 Introduction

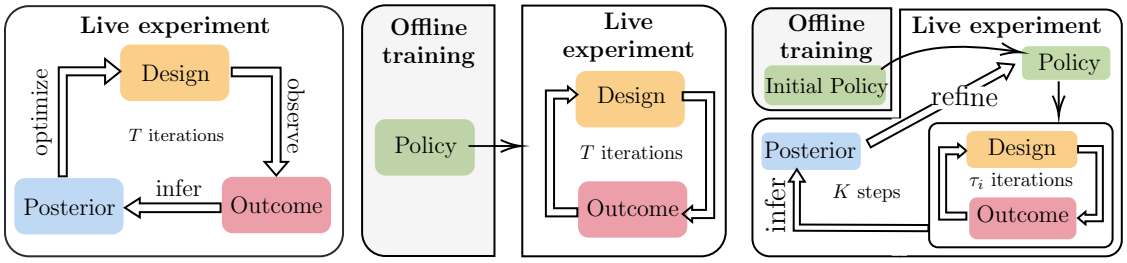
Adaptive experimentation plays a crucial role in science and engineering: it enables targeted and efficient data acquisition by sequentially integrating information gathered from past experiment iterations into subsequent design decisions (MacKay, 1992; Atkinson et al., 2007; Myung et al., 2013). For example, consider an online survey that aims to infer individual preferences through personalized questions. By strategically tailoring future questions based on insights from past responses, the survey can rapidly hone in on relevant questions for each specific individual, enabling precise preference inference with fewer, more targeted questions.

Bayesian experimental design (BED) offers a principled mathematical framework for solving the adaptive design problem (Chaloner and Verdinelli, 1995; Ryan et al., 2016; Rainforth et al., 2023). In the BED framework, the quantity of interest (e.g. individual preferences), is represented as an unknown parameter θ and modelled probabilistically through a joint generative model on θ and experiment outcomes given designs. The goal is then to choose designs that are maximally informative about θ. Namely, we maximize the *Expected Information Gain* (EIG, Lindley, 1956, 1972), which measures the expected reduction in our uncertainty about θ from running an experiment with a given design.

The *traditional* adaptive BED approach (Fig 1a) involves iterating between making design decisions by optimizing the EIG of the next experiment step, and updating the underlying model through Bayesian inference, conditioning on data obtained so far. Unfortunately, this approach leads to sub-optimal design decisions, as it is a greedy, myopic, strategy that fails to plan for future experiment steps (Huan and Marzouk, 2016; Foster, 2021). Furthermore, it requires substantial computation at each experiment iteration (posterior update and EIG optimization), making it impractical for real-time applications (Rainforth et al., 2023).

Foster et al. (2021) showed that this traditional framework can be significantly improved upon by instead taking a *policy-based* BED (PB-BED) approach. As shown in Fig 1b, their Deep Adaptive Design (DAD) framework, and its subsequent extensions (Ivanova et al., 2021; Blau et al., 2022; Lim et al., 2022), are based on learning a *design policy network* upfront, mapping experimental histories to next design. This *fully amortized* approach

 [†]Work undertaken whilst at the University of Oxford. Correspondence to: {desi.ivanova, marcel.hedman}@stats.ox.ac.uk .



(a) Traditional adaptive BED (b) Fully-amortized PB-BED (c) Semi-amortized PB-BED

Figure 1: **Overview of adaptive BED approaches.** The traditional BED approach fits a posterior after each experiment iteration and optimizes for the next step best designs (i.e. greedily). Fully amortized policy-based BED approaches like DAD train a policy once offline, before the live experiment, then deploy this as a fixed policy to make adaptive design decisions during the experiment itself. Our proposed semi-amortized PB-BED approach enables periodic policy refinement, assimilating data from the real-world live experiment.

avoids any significant computation during the experiment, enabling real-time, adaptive, and non-myopic design strategies that represent the current state-of-the-art in adaptive BED.

In principle, fully amortized approaches can learn optimal design strategies (in terms of total EIG). In practice, learning a policy that remains optimal for all possible experiment realizations is rarely realistic. In particular, the dimensionality of experimental history expands as the experiment progresses, making it increasingly difficult to account for all possible eventualities through upfront training alone. Moreover, deficiencies in our model means that observed data can be distinct from the simulated one used to train the policy.

To address these limitations, and allow utilisation of any computation that is available during the experiment, we introduce a hybrid, *semi-amortized* PB-BED approach, called *Step-wise Deep Adaptive Design* (Step-DAD). As illustrated in Fig 1c, Step-DAD periodically updates the policy during the experiment. This allows the policy itself to be adapted based on previously gathered data, refining it to maximize performance for the particular realization of the data that we are observing. In turn, this allows Step-DAD to make more accurate design decisions and provides significant improvements in robustness to observing data that is dissimilar to that generated in the original policy training. Our empirical evaluations reveal that Step-DAD is able to provide substantial improvements in state-of-the-art design performance, while still using substantially less computation than traditional adaptive BED.

2 Background

Guided by the principle of information maximization, Bayesian experimental design (BED, Lindley, 1956) is a model-based framework for designing optimal experiments. Given a model $p(\theta)p(y \mid \theta, \xi)$, describing the relationship between experiment outcomes y, controllable designs ξ and parameters of interest θ, the goal is to select the design ξ that maximizes the expected information gain (EIG) about θ. The EIG (equivalent to mutual information) is the expected reduction in Shannon entropy from the prior to the posterior distribution of θ:

$$I(\xi, y) = \mathbb{E}_{p(y|\xi)}[H[p(\theta)] - H[p(\theta \mid \xi, y)]], \quad \text{where } p(y \mid \xi) = \mathbb{E}_{p(\theta)}[p(y \mid \theta, \xi)], \tag{1}$$

In the following subsections, we highlight works most closely related to ours, with a detailed discussion available in Section 5.

2.1 Traditional Adaptive BED

BED becomes particularly powerful in adaptive experimental contexts, where we allow the next design, ξ_t, to be informed by the data acquired up to that point, $h_{t-1} \coloneqq (\xi_1, y_1), \ldots, (\xi_{t-1}, y_{t-1})$, which we refer to as the *history*. In the traditional adaptive BED framework (Ryan et al., 2016), this is done by assimilating the data into the model by fitting the posterior $p(\theta \,|\, h_{t-1})$, followed by the maximizing the one-step ahead or *incremental* EIG

$$I^{h_{t-1}}(\xi_t) = \mathbb{E}_{p(y|\xi_t, h_{t-1})}\big[H[p(\theta \,|\, h_{t-1})] - H[p(\theta \,|\, h_t)]\big], \tag{2}$$

where $p(y \,|\, \xi_t, h_{t-1}) = \mathbb{E}_{p(\theta \,|\, h_{t-1})}[p(y \,|\, \theta, \xi_t, h_{t-1})]$. We use the superscript h_{t-1} to emphasize conditioning on the history currently available, setting $h_0 = \varnothing$.

Whilst this traditional framework offers a principled way to optimize experimental designs, it comes with some limitations. One drawback is its myopic nature that greedily maximizes for the next best design and overlooks the impact of future experiments, ultimately leading to sub-optimal design decisions. Another limitation is the significant computational expense incurred from the iterative posterior inference and EIG optimization. In general, the posterior computation is intractable and the EIG (2) estimation is *doubly intractable* (Rainforth et al., 2018; Foster et al., 2019). Since these steps must be conducted at each experiment iteration, the traditional adaptive BED approach is often impractical for real-time applications.

2.2 Amortized Policy-Based BED

In response to the limitations of traditional adaptive BED, Foster et al. (2021) introduce the idea of amortizing the adaptive design process through learnt policies. This amortized policy-based BED (PB-BED) approach represents a significant advancement over the traditional framework, delivering state-of-the-art non-myopic design optimization whilst enabling real-time deployment. PB-BED reformulates the design problem using a policy π, which maps experimental histories to next design, $\pi : h_{t-1} \mapsto \xi_t$. The optimal policy maximizes the *total* EIG across the T experiments (Foster et al., 2021; Shen and Huan, 2021)

$$\mathcal{I}_{1\to T}(\pi) = \mathbb{E}_{p(h_T|\pi)}\left[H[p(\theta)] - H[p(\theta \,|\, h_T)]\right], \tag{3}$$

$$= \mathbb{E}_{p(\theta)p(h_T \,|\, \theta, \pi)}\left[\log p(h_T \,|\, \theta, \pi) - \log p(h_T \,|\, \pi)\right] \tag{4}$$

where $p(h_T \,|\, \theta, \pi) = \prod_{t=1}^{T} p(y_t \,|\, \theta, \xi_t, h_{t-1})$, $p(h_T \,|\, \pi) = \mathbb{E}_{p(\theta)}[p(h_T \,|\, \theta, \pi)]$, and $\xi_t = \pi(h_{t-1})$ are all evaluated autoregressively. This policy-based formulation strictly generalizes the traditional adaptive BED approach, which can be viewed as learning a policy that maximizes the incremental one-step-ahead EIG (2) at each iteration, $\pi(h_{t-1}) = \arg\max_{\xi_t} I^{h_{t-1}}_{t-1\to t}(\xi_t)$.

Whilst the total EIG formulation (3) provides a unified training objective for the policy, it remains doubly intractable like the standard EIG. The original Deep Adaptive Design (DAD) of Foster et al. (2021) addressed this by using tractable variational lower bounds of the EIG (Foster et al., 2019, 2020; Kleinegesse and Gutmann, 2020) coupled with stochastic gradient ascent (SGA) schemes to directly train a policy network taking the form of a neural network directly mapping from histories to design decisions. It thus provided a foundation for conducting PB-BED in practice. A number of extensions to the DAD

approach have since been developed, e.g. (Ivanova et al., 2021; Blau et al., 2022; Lim et al., 2022), broadening its applicability to a wider class of models by proposing alternative policy training schemes. All share a core methodology, where the policy network is trained only once, offline, with hypothetical experimental histories simulated from the assumed generative model $p(\theta)p(h_T \mid \theta, \pi)$. Once trained, it remains unchanged during the live experiment and across multiple experimental instances (e.g. different survey participants), as illustrated in Fig 1b. This *fully amortized* approach eliminates the need for posterior inference and EIG optimization at each experiment iteration, thereby enabling almost instant design decisions.

3 Semi-Amortized PB-BED

Fully amortized PB-BED methods enable real-time deployment and provide design decisions that are typically superior to those of the traditional framework. However, there are many problems where we can afford to perform some computation during the experiment itself. It is therefore natural to ask whether we can usefully exploit such computational availability to further improve the quality of our design decisions? In particular, the fact that the current state-of-the-art approaches for design quality are all fully amortized suggests that improvements should be possible when this is not a computational necessity.

To address this, we note that the computational gains of fully amortized PB-BED methods come at the cost of their inability to adapt the *policy itself* in response to acquired experimental data. We argue that this rigidity leads to sub-optimal designs decisions, particularly in scenarios where real-world experimental data significantly deviates from the simulated one used during training of the policy. Two main factors contribute to this issue:

Imperfect training In fully amortized PB-BED we simulate experimental histories to try and learn a policy that will generalize across the entire experimental space—effectively learning a regressor from all possible histories to design decisions. However, the effectiveness of any learner with finite data/training is inevitably limited, especially in regions of the input space where training data is sparse. In short, we are learning a policy to cover all possible histories we might see, but at deployment we are dealing only with a specific history that may be similar to few, if any, of the histories with simulated during training. This challenge is particularly exacerbated in experiments with extended horizons, due to the high dimensionality of the resulting histories. Additionally, the finite representational capacity of the policy hinders perfect approximation even with infinite data. Together these lead to a discrepancy between the learned policy π and the true optimal design strategy π^*, producing an **approximation gap** for the learned policies.

Double reliance on the generative model Fully amortized PB-BED relies on the generative model to both simulate experimental histories for policy training and to evaluate the success of our design decisions via the total resulting information gained. In other words, we use the model in both the expectation and information gain elements of the EIG in (3). This dual reliance magnifies the consequences of model misspecifications (Overstall and McGree, 2022; Go and Isaac, 2022). Moreover, even if the model is well-specified from a Bayesian inference perspective (Uppal and Wang, 2003), there might still be significant discrepancies between the prior-predictive distribution, $p(h_T|\pi)$, used to simulate data in the policy training and the true underlying data generating distribution.

The upshot of this is that we may see data at deployment that is highly distinct to any of that simulated during the policy training. The lack of mechanisms for integrating real experimental data into the policy means that fully amortized approaches have no mechanism to overcome this issue. This can be characterized as a form of **generalization gap**—the learned policy fails to generalize to the real-world experimental conditions, caused by its inability to integrate and respond to the actual experimental data gathered so far.

3.1 Online policy updating

To address these limitations, we propose a *semi-amortized* PB-BED framework, which introduces dynamic adaptability by allowing periodic updates to the policy during deployment in response to acquired experimental data. The core idea behind our semi-amortized approach is based on the intuition that whilst a fully amortized policy is a strong starting point, it can be significantly enhanced through targeted refinements during the experiment. Focusing first on the case of a single policy update, the following proposition formalizes this intuition and lays the theoretical foundation for semi-amortized PB-BED.

Proposition 1 (Decomposition of total EIG)**.** *For any design policy π, the total EIG of a T-step experiment can be decomposed as $\mathcal{I}_{1\to T}(\pi) = \mathcal{I}_{1\to\tau}(\pi) + \mathbb{E}_{p(h_\tau \mid \pi)}[\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi)]$, for any intermediate step $1 \leq \tau \leq T$, where*

$$\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi) = \mathbb{E}_{p(\theta|h_\tau)p(h_{\tau+1:T} \mid h_\tau,\theta,\pi)}\left[\log \frac{p(h_{\tau+1:T} \mid h_\tau, \theta, \pi)}{p(h_{\tau+1:T} \mid h_\tau, \pi)}\right]. \tag{5}$$

Proof is given in Appendix B. The decomposition of the total EIG into two distinct components—the EIG accumulated up to an intermediate step τ, and the expected EIG for subsequent steps conditional on the history h_τ gathered until that point—highlights a crucial aspect of our semi-amortized approach: that the optimality of a policy for the later phases of the experiment, from step $\tau + 1$ to T, is solely determined by the model and already collected data h_τ. That is, it is independent of the policy deployed during the first τ experiments. As such, we can use a strategy where, regardless of the initial policy performance, we can construct a new policy at some intermediate step τ that optimizes future remaining experimental designs, without being constrained by past decisions.

Our semi-amortized approach is now based around exploiting this flexibility to refine the policy midway through the experiment by introducing a *step design policy* $\pi^{s(\tau)}$. Initially, $\pi^{s(\tau)}$ uses the fully amortized policy π_{h_0} for the first τ steps of the experiment. After step τ, it switches to a new policy $\pi_{h\tau}$, trained to maximize the total *remaining* EIG, $\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi)$ (5).

This gives us an **infer-refine** process for semi-amortization in PB-BED that mirrors the two stage procedure characteristic of traditional adaptive BED (cf Fig. 1a and Fig. 1c). The **infer** stage entails fitting the posterior distribution $p(\theta \mid h_\tau)$ with the data up to τ. The subsequent **refine** stage learns a customized policy π_{h_τ} for the remaining steps of the experiment by maximizing (5). It therefore allows for more effective design decisions than the fully amortized approach. However, unlike the traditional BED approach, which is greedy and requires updates at every experimental step, our semi-amortized method offers a superior non-myopic design strategy and allows for selective updates.

We acknowledge that this approach requires some computation to be performed during the live experiment, which can pose challenges in applications where design decisions must be made very quickly. However, in many applications there is computation time available,

and our semi-amortized PB-BED approach can exploit this, even if that time is limited. As we show in subsequent sections, significant improvements to the policy can often be achieved with minimal additional training, such that substantial gains are often possible without drastically compromising deployment speed.

Multi-step policy updates We can naturally extend our approach to include a multi-step update mechanism, enabling a more dynamic and responsive policy adaptation over the course of the live experiment. To this end, we define a *refinement schedule*, $\mathcal{T} = \tau_0, \tau_1, \cdots, \tau_K$— an increasing sequence defining the points at which the policy is refined. We adopt the convention $\tau_0 = 0$ and $h_0 = \varnothing$, marking the offline optimization of the fully-amortized policy π_{h_0}. For $\tau_k > 0$, we follow our two-stage infer-refine procedure.

4 Step-Wise Deep Adaptive Design

We introduce **Step-Wise Deep Adaptive Design** (Step-DAD) as a way of implementing the semi-amortized PB-BED in practice. Building on DAD and the infer-refine procedure outlined in the last section, Step-DAD employs stochastic gradient ascent schemes to optimize variational lower bounds on the remaining EIG (5) to sequentially train the step policy $\pi^{s(\mathcal{T})}$ in a scalable manner. An overview of Step-DAD is presented in Algorithm 1 in the Appendix.

The two key components of Step-DAD's aforementioned infer-refine procedure are an inference method for approximating $p(\theta|h_\tau)$, and a refinement strategy for using this to update our policy. Standard inference techniques (such as variational inference and Monte Carlo methods) are used for the former as discussed in our experiments. Our focus here will instead be on our specialized procedure for policy refinement and the policy architecture itself.

4.1 Policy refinement

Due to its doubly intractable nature, the task of optimizing the remaining EIG, $\mathcal{I}^{h_{\tau_k}}_{\tau_k+1\to T}(\pi)$, presents a notable challenge (Rainforth et al., 2018; Foster et al., 2019). In selecting an appropriate scalable and computationally tractable estimator for it, we wish to ensure compatibility with a wide range of inference schemes for $p(\theta \,|\, h_{\tau_k})$. Namely, as this serves as an updated 'prior' during the policy refinement, it is important that we use an EIG estimator that does not require evaluations of the prior density, to ensure compatibility with sampling-based inference schemes.

Lower bound estimators such as the explicit-likelihood-based sequential Prior Contrastive Estimator (sPCE, Foster et al., 2021), as well as the implicit likelihood InfoNCE (van den Oord et al., 2018; Ivanova et al., 2021) and NWJ (Nguyen et al., 2010; Kleinegesse and Gutmann, 2020) bounds, align with this requirement. For generative models with explicit likelihoods (implicit models are discussed in Appendix C), we, therefore, choose to employ the sPCE lower bound, defined as

$$\mathcal{L}^{\text{sPCE}}_{\tau_k+1\to T}(\pi) = \mathbb{E}\left[\log \frac{p(h_{\tau_k+1:T} \,|\, h_{\tau_k}\theta_0, \pi)}{\frac{1}{L+1}\sum_{\ell=0}^{L} p(h_{\tau_k+1:T} \mid \theta_\ell, \pi)}\right]. \tag{6}$$

Here, the expectation is taken over a 'positive' prior sample $\theta_0 \sim p(\theta_0 \,|\, h_{\tau_k})$, future design-outcome pairs under it $h_{\tau_k+1:T} \sim \prod_{t=\tau_k+1}^{T} p(h_t \,|\, h_{\tau_k}\theta_0, \xi_t)$, $\xi_t = \pi(h_{t-1})$, and L 'contrastive' prior samples $\theta_{1:L} \sim \prod_{\ell=1}^{L} p(\theta_\ell \,|\, h_{\tau_k})$. Step-DAD parameterizes π by a neural network and optimizes an appropriate objective, such as $\mathcal{L}^{\text{sPCE}}_{\tau_k+1\to T}(\pi)$, with respect to the network

parameters using standard stochastic gradient ascent (SGA) schemes (Robbins and Monro, 1951; Kingma and Ba, 2014). Following (Foster et al., 2021), we use path-wise gradients in the case of reparametrizable distributions (Rezende et al., 2014; Mohamed et al., 2020), and score function (REINFORCE) otherwise (Williams, 1992).

4.2 Policy architecture

It is possible to optimize (6) by training an entirely new policy $\pi_{h_{\tau_k}}$ at each refinement step τ_k, allowing flexibility to even select different architectures. Though such a strategy may occasionally be advantageous, we instead, we propose a more pragmatic and lightweight approach: leveraging the already established fully amortized policy π_{h_0} as a baseline and fine-tuning it for subsequent steps. Though our experiments perform a full fine-tuning of all policy parameters, it is also possible to implement more parameter-efficient methods.

For the policy architecture, similar to (Foster et al., 2021), we individually embed each design-outcome pair $(\xi_i, y_i) \in h_t$ into a fixed-dimensional representation before aggregating them across t into a summary vector. This allows condensing varied-length experimental histories into a consistent dimensionality, enabling the handling of variable history sizes. Finally, the summary vector is then mapped to the next experimental design ξ_{t+1}. For the aggregation mechanism, the choice between permutation invariant and autoregressive architectures depends on the nature of the data. When the data h_t is exchangeable, permutation invariant architectures like DeepSets (Zaheer et al., 2017) or SetTransformer (Lee et al., 2019) are suitable. In contrast, sequential or time-series data would benefit from autoregressive models like transformers (Vaswani et al., 2017).

5 Related Work

The idea of using a design policy in the context of adaptive BED was first proposed by Huan and Marzouk (2016). Leveraging dynamic programming principles, the policy they learn aims to establish a mapping from explicit posterior representations—serving as the *state* in reinforcement learning (RL) terminology—to subsequent design choices. As a result, each iteration of the experiment necessitates substantial computational resources for updating the posterior. The concept of fully amortized policy-based BED, which directly maps data collected to design decisions, has only recently been introduced (Foster et al., 2021) and subsequently extended to differentiable implicit models (Ivanova et al., 2021). RL algorithms have also been employed to optimize design policies (Blau et al., 2022; Lim et al., 2022). None of the previous approaches have looked to refine the policy during the experiment itself.

As discussed in § 2.1, adaptive BED has traditionally employed a two-step greedy strategy, involving EIG optimization and posterior inference optimization. For **EIG estimation** established methods include nested Monte Carlo (Myung et al., 2013; Vincent and Rainforth, 2017), variational bounds (Foster et al., 2019, 2020; Kleinegesse and Gutmann, 2020) and ratio estimation (Kleinegesse and Gutmann, 2019; Kleinegesse et al., 2021). The subsequent **EIG maximization** has historically relied on gradient-free optimization, including grid-search, evolutionary algorithms (Price et al., 2018), Bayesian optimization (Kleinegesse et al., 2021; Foster et al., 2019), or Gaussian process surrogates (Overstall and McGree, 2020). Thanks to advancements in gradient-based methods, EIG estimation and optimization can be performed jointly in a SGA scheme (Huan and Marzouk, 2014; Foster et al., 2020; Kleinegesse and Gutmann, 2021).

Method	Lower bound (↑)	Upper bound (↑)
Random	3.612 ± 0.012	3.613 ± 0.012
Static	3.945 ± 0.026	3.946 ± 0.026
Step-Static	3.974 ± 0.008	3.975 ± 0.008
DAD	6.771 ± 0.012	6.803 ± 0.013
Step-DAD	**7.605 ± 0.078**	**7.609 ± 0.078**

Table 1: **Source Location Finding.** For Step-DAD we report best finetuning step, $\tau = 6$, and the rest in Table 8.

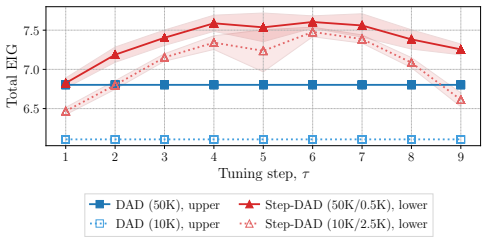


Figure 2: **Source location finding: Sensitivity to training budget.**

The **inference scheme** used for model updates in the traditional BED framework is often contingent on the availability of a closed-form likelihood. In cases where it is available, Monte Carlo based methods are typically used, such as Sequential Monte Carlo (Del Moral et al., 2006; Drovandi et al., 2014) or population Monte Carlo (Rainforth, 2017). In likelihood-free settings, techniques like approximate Bayesian computation (ABC) (Lintusaari et al., 2017; Sisson et al., 2018), ratio estimation (Thomas et al., 2016), or approximating the likelihood first, e.g. via polynomial chaos expansion (Huan and Marzouk, 2013), are typically utilized. In addition to the presence of a likelihood function, practical considerations influencing the choice of posterior inference methods include a trade-off between speed and accuracy, weighing options like (amortized) variational inference (Zhang et al., 2018) against asymptotically exact, but slow MCMC methods such as HMC (Betancourt, 2017).

The challenge of **model misspecification** in BED remains a relatively underexplored area, with foundational insights provided by (Farquhar et al., 2021) and (Overstall and McGree, 2022). Fully amortized PB-BED is particularly vulnerable to model misspecification due to its reliance on a singular learning phase without the capacity to integrate real-world experimental feedback. Addressing these challenges in the literature is limited, with some approaches recommending the adoption of a more robust EIG objective as a potential solution (Go and Isaac, 2022). Our semi-amortized PB-BED methodology, whilst not directly tackling the issue of misspecification, inherently enhances robustness by enabling iterative data integration and policy refinement.

6 Experiments

We empirically evaluate our semi-amortized **Step-DAD** approach on a range of design problems, comparing its performance against **DAD** to determine the additional EIG achieved by $\pi^{s(\mathcal{T})}$ over π_{h_0}. Full details about the models and baselines are provided in Appendix D.

6.1 Source Location Finding

We consider the source location finding experiment from Foster et al. (2021), which draws upon the acoustic energy attenuation model detailed in Sheng and Hu (2005). The objective of the experiment is to infer the locations of some hidden sources from noisy measurements, y, of their combined signal intensity. Each source emits a signal whose intensity diminishes following the inverse-square law relative to distance. Full model details are in Appendix D.4.

We begin by learning a fully amortized DAD policy to perform $T = 10$ experiments to locate a single source. A training budget of 50K gradient steps was employed for this policy, as it was found that further training did not significantly improve performance for our chosen architecture. We run Step-DAD for $\tau = 1, \ldots, 9$, using importance sampling

θ dim	EIG difference	DAD, total EIG
4	0.701 ± 0.023	6.483 ± 0.055
8	0.426 ± 0.014	7.111 ± 0.067
12	0.423 ± 0.012	6.956 ± 0.056

Table 2: **Locating multiple sources**.

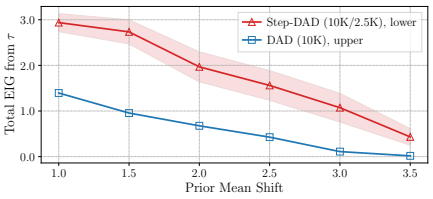


Figure 3: **Sensitivity to prior perturbations.**

to approximate the posterior $p(\theta \mid h_\tau)$. Results in Table 1 highlight the best performing finetuning step, $\tau = 6$, at which Step-DAD surpasses all baseline methods. Table 8 in the Appendix shows that $\tau = 4, 5, 7$ exhibit performance statistically equivalent to that of $\tau = 6$.

Sensitivity to training budget We investigate the overall resource efficiency of Step-DAD by comparing to DAD under two training regimes. Concretely, we consider two budget levels for the training of the fully amortized policy: *full* pre-training budget of 50K gradient steps, and *reduced* pre-training budget of 10K steps (5× lower), with a subsequent fine-tuning of 0.5K and 2.5K steps, respectively. Figure 2 presents a conservative comparison between the two methods by showing upper bound estimates for DAD and lower bound estimates for Step-DAD. The findings illustrate that Step-DAD surpasses the corresponding DAD baseline across both budget levels. The Step-DAD variant that starts with a modest pre-training budget of 10K steps, followed by a fine-tuning phase of 2.5K steps, consistently outperforms the DAD model trained on a 50K step budget, except at the boundary tuning steps $\tau = 1$ and $\tau = 9$. The performance advantage of Step-DAD is most pronounced when fine-tuning occurs midway through the experiment ($\tau = 5, 6$), where our method can effectively leverage the accumulated data to refine the policy and have enough experiments remaining to usefully deploy the improved customized policy.

Scaling up In the more complex setting of locating multiple sources, our method demonstrates strong performance gains compared to DAD. We consider $2, 4$ and 6 sources, resulting in 4-, 8- and 12-dimensional unknown parameter, respectively. Table 2 highlights the scalability of our method to higher-dimensional parameter spaces.

Robustness to prior perturbations We evaluate the robustness of Step-DAD when the prior distribution at test time diverges from that used during the offline training phase. This scenario mirrors real-world situations where the conditions during the live experiment may not match those assumed when training the fully amortized policy. We consider total EIG **under an alternative model**, $\mathcal{I}_{\tilde{p}(\theta)}(\pi) \coloneqq \mathbb{E}_{\tilde{p}(\theta)p(h_T|\theta,\pi)}\left[\log \frac{p(h_T \mid \theta,\pi)}{p(h_T \mid \pi)}\right]$, where $p(h_T \mid \pi) = \mathbb{E}_{\tilde{p}(\theta)}[p(h_T \mid \theta, \pi)]$ and $\tilde{p}(\theta)$ is the perturbed prior distribution.

Results are shown in Figure 3. The EIG for DAD decreases to essentially zero with the increased prior shift, whilst Step-DAD is still able to deliver positive information gains. This robustness is anticipated due to Step-DAD's ability to assimilate the data gathered and make policy adjustments in light of new evidence, which is crucial for maintaining an informative design strategy under uncertain model assumptions.

6.2 Hyperbolic Temporal Discounting

Temporal discounting describes the tendency for individuals to prefer for smaller immediate rewards over larger delayed ones. This phenomenon is a key concept in psychology and

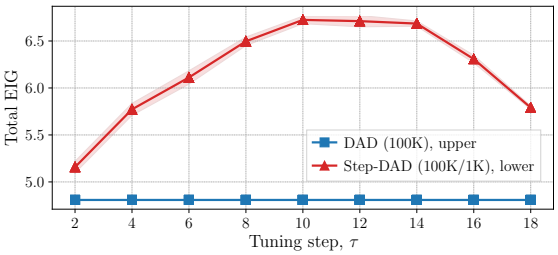


Figure 4: **Temporal discounting model.** EIG improvement of Step-DAD over DAD after fine-tuning the policy at step τ. The DAD network is trained for 100K steps, subsequent policy refinement for 1K steps.

Method	Upper bound (↑)	Lower bound (↑)
Kirby (2009)	1.861 ± 0.008	1.864 ± 0.009
Static	2.518 ± 0.007	2.524 ± 0.007
Frye et al. (2016)	3.500 ± 0.029	3.513 ± 0.029
BADapted	4.454 ± 0.016	4.536 ± 0.018
DAD	4.778 ± 0.013	4.808 ± 0.014
Step-DAD (τ=10)	**6.711 ± 0.040**	**6.721 ± 0.040**

Table 3: **Temporal discounting.** Upper and lower bound estimates of total EIG. Errors show ± 1s.e., over 16 (2048) histories for step methods (rest). All baselines except DAD as reported in Foster et al. (2021).

economics and has been widely applied to study important social and individual behaviors (Critchfield and Kollins, 2001), including dietary choices (Bickel et al., 2021; McClelland et al., 2016), exercise habits (Tate et al., 2015), patterns of substance abuse and other unhealthy behaviour (Holt et al., 2003; Story et al., 2014).

An individual's time delay preference is typically inferred by asking a series of questions "Would you prefer $\$R$ now or \$100 in D days time?" The tuple $\xi = (R, D)$ defines our experimental design, and the the outcome y is the participant's decision to *accept* or *reject* the delay. For example, a participant might prefer an immediate $R = \$90$ over \$100 in $D = 30$ days, but might choose differently if the delay was shortened to $D = 7$ days.

Single update Using the hyperbolic discounting model introduced in Mazur (1987) and as implemented by Vincent (2016), we train DAD policy for 100K gradient steps, aimed at designing $T = 20$ experiments. We select a grid of tuning steps τ in the range from 2 to 18 in increments of 2. For posterior inference, we use simple importance sampling to draw samples from the posterior $p(\theta \,|\, h_\tau)$ and 1% of the training budget (i.e. 1K gradient steps).

Figure 4 reports the results and illustrates that our method yields an improvement in total EIG *across all tuning steps* τ when compared to the baseline DAD policy. The maximal increase occurs around the middle of the experiment, aligning with intuition: at this point, sufficient data has been accumulated to inform a meaningful posterior update, whilst sufficient number of experiments remain to effectively deploy the refined policy. Table 3 demonstrates the superiority of Step-DAD over conventional baselines, including those derived from psychology research (Kirby, 2009; Frye et al., 2016; Vincent and Rainforth, 2017) and traditional BED approaches such as BADapted (Vincent and Rainforth, 2017). It also outperforms the static BED strategy, highlighting the effectiveness of adaptive design strategies in extracting more valuable information from experiments.

Multiple updates and design extrapolation We extend the deployment of DAD and Step-DAD to $T = 40$ experiments, doubling the scope at which they were originally trained, i.e. without retraining the DAD network. Step-DAD is fine-tuned at two steps, τ and 2τ, with $\tau \in \{5, 6, 7, 8\}$. As Table 4 shows, Step-DAD demonstrates significantly improved capacity to extract information in the later stages of the experiment, beyond its initial training.

	EIG from τ (↑)		EIG from 2τ (↑)	
τ	DAD	Step-DAD	DAD	Step-DAD
5	3.9 ± 0.24	$\mathbf{4.8 \pm 0.14}$	2.1 ± 0.36	$\mathbf{4.6 \pm 0.18}$
6	3.4 ± 0.30	$\mathbf{5.1 \pm 0.07}$	1.7 ± 0.30	$\mathbf{4.7 \pm 0.12}$
7	3.0 ± 0.35	$\mathbf{4.4 \pm 0.30}$	1.3 ± 0.23	$\mathbf{4.4 \pm 0.11}$
8	2.6 ± 0.36	$\mathbf{4.7 \pm 0.13}$	1.0 ± 0.19	$\mathbf{4.2 \pm 0.13}$

Table 4: **Hyperbolic temporal discounting: extrapolating designs.** Comparison of EIG upper bound for DAD and lower bound for Step-DAD across different tuning steps τ with $T = 40$. Errorbars indicate $\pm$1s.e. computed over 16 histories.

7 Conclusions

In this work, we introduced the idea of a semi-amortized approach to PB-BED that enhances the flexibility, robustness and effectiveness of fully amortized design polices. Our method, Step-wise Deep Adaptive Design (Step-DAD), dynamically updates its step policy in response to new data through a systematic 'infer-refine' procedure that refines the design strategy for the remaining experiments in light of the experimental data gathered so far. This iterative refinement enables the step policy to evolve as the experiment progresses, ensuring more robust and tailored design decisions, as demonstrated in our empirical evaluation. Step-DAD marks a step forward in our ability to conduct more efficient, informed, and robust experiments, opening new avenues for exploration in various scientific domains.

Acknowledgements

DRI is supported by EPSRC through the Modern Statistics and Statistical Machine Learning (StatML) CDT programme, grant no. EP/S023151/1. MH is supported by funding provided by Novo Nordisk.

Impact Statement

While the deployment of adaptive data acquisition techniques must always be carefully considered to avoid imparting biases or other negative effects, the focus of this work is on purely methodological advancements and we do not envision any direct potential negative societal consequences relative to previous work in the area.

Reproducibility Statement

All our experiments are conducted using synthetic data. Code will be made publicly available.

Appendix A. Algorithm

Algorithm 1: Overview of Step-DAD

Input: Generative model $p(\theta)p(y \mid \theta, \xi)$, experimental budget T, refinement schedule $\mathcal{T}=\{\tau_0, \tau_1, \ldots, \tau_{K+1}\}$, with $\tau_0{=}0, \tau_{K+1}{=}T$, training budgets $\{N_{\tau_k}\}_{k=1:K}$

Output: Dataset $h_T = \{(\xi_t, y_t)\}_{t=1:T}$

Offline stage: Before the live experiment

▷ Set $h_0 = \varnothing$.

while Computational budget does not exceed N_0 **do**

▷ Train fully-amortized π_{h_0} as in Foster et al. (2021)

end

Online stage: During the live experiment

for $k = 1, \ldots, K+1$ **do**

for $\tau_{k-1} < t \leq \tau_k$ **do**

▷ Compute design $\xi_t = \pi_{h_{\tau_{k-1}}}(h_t)$

▷ Run experiment ξ_t, observe an outcome y_t

▷ Update the dataset $h_t = h_{t-1} \cup (\xi_t, y_t)$

end

If $k = K+1$ **then return** h_T **end**

while Computational budget does not exceed N_k **do**

▷ Fit a posterior $p(\theta \mid h_{\tau_k})$

▷ Fine-tune policy $\pi_{h_{\tau_k}}$ by optimizing (6)

end

end

Appendix B. Proofs

Proposition 1 (Decomposition of total EIG)**.** *For any design policy π, the total EIG of a T-step experiment can be decomposed as $\mathcal{I}_{1\to T}(\pi) = \mathcal{I}_{1\to\tau}(\pi) + \mathbb{E}_{p(h_\tau \mid \pi)}[\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi)]$, for any intermediate step $1 \leq \tau \leq T$, where*

$$\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi) = \mathbb{E}_{p(\theta|h_\tau)p(h_{\tau+1:T} \mid h_\tau, \theta, \pi)}\left[\log \frac{p(h_{\tau+1:T} \mid h_\tau, \theta, \pi)}{p(h_{\tau+1:T} \mid h_\tau, \pi)}\right]. \tag{5}$$

Proof We can write the likelihood and marginal as

$$p(h_T \mid \theta, \pi) = p(h_\tau \mid \theta, \pi)p(h_{\tau+1:T} \mid h_\tau, \theta, \pi), \quad p(h_T \mid \pi) = p(h_\tau \mid \pi)p(h_{\tau+1:T} \mid h_\tau, \pi)$$

Substituting in the definition of total EIG (3) and rearranging

$$\begin{aligned}\mathcal{I}_{1\to T}(\pi) &= \mathbb{E}_{p(\theta)p(h_\tau|\theta,\pi)}\left[\log \frac{p(h_\tau \mid \theta, \pi)}{p(h_\tau \mid \pi)}\right] + \\ &\qquad \mathbb{E}_{p(h_\tau|\pi)p(\theta|h_\tau)p(h_{\tau+1:T}|h_\tau,\theta,\pi)}\left[\log \frac{p(h_{\tau+1:T} \mid h_\tau, \theta, \pi)}{p(h_{\tau+1:T} \mid h_\tau, \pi)}\right] \\ &= \mathcal{I}_{1\to\tau}(\pi) + \mathbb{E}_{p(h_\tau \mid \pi)}[\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi)]\end{aligned}$$

as required. ■

Appendix C. Further EIG bounds

The sequential Nested Monte Carlo (sNMC) (Foster et al., 2021) upper bound is given by

$$\mathcal{U}^{\text{sNMC}}_{\tau_k+1\to T}(\pi) := \mathbb{E}\left[\log \frac{p(h_{\tau_k+1:T} \mid h_{\tau_k}\theta_0, \pi)}{\frac{1}{L}\sum_{\ell=1}^{L} p(h_{\tau_k+1:T} \mid \theta_\ell, \pi)}\right], \tag{7}$$

which we use to evaluate different design strategies.

For implicit models we can utilize the InfoNCE bound (van den Oord et al., 2018), which is given by

$$\mathcal{L}_{\text{InfoNCE}}(\pi, U; L) := \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)}\mathbb{E}_{p(\theta_{1:L})}\left[\log \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1}\sum_{i=0}^{L}\exp(U(h_T, \theta_i))}\right], \tag{8}$$

or the NWJ bound (Nguyen et al., 2010), given by:

$$\mathcal{L}_{\text{NWJ}}(\pi, U) := \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)}\left[U(h_T, \theta) - e^{-1}\mathbb{E}_{p(\theta)p(h_T|\pi)}\left[\exp(U(h_T, \theta))\right]\right] \tag{9}$$

where, for both bounds, U is a learnt critic function, $U : h_T \times \theta \mapsto \mathbb{R}$.

Appendix D. Experiment details

D.1 Computational resources

The experiments were conducted using Python and open-source tools. PyTorch (Paszke et al., 2019) and Pyro (Bingham et al., 2018) were employed to implement all estimators and models. Additionally, MlFlow (Zaharia et al., 2018) was utilized for experiment tracking and management. Experiments were performed on two separate GPU servers, one with 4xGeForce RTX 3090 cards and 40 cpu cores; the other one with 10xA40 and 52 cpu cores. Every experiment was run on a single GPU.

D.2 Evaluation details

In addition to DAD, we consider several other baseline strategies for comparison. **Static** design learns all T designs prior to the experiment and remain fixed throughout it. We consider a **Step-Static** approach where, akin to a semi-amortized static method, at step τ, we randomly select τ designs from the total of T and retrain the $T - \tau$ remaining ones. If appropriate, we include a **Random** design strategy serving as a naive, non-optimized benchmark, and **problem-specific** baselines if available.

Our main metric for assessing the quality of various design strategies is the **total EIG**, $\mathcal{I}_{1\to T}(\pi)$, as given in (3). We approximate it via the sPCE lower bound (6) along with its upper bound counterpart—the sequential Nested Monte Carlo estimator (sNMC, Foster et al., 2021). For the remaining EIG (5), we leverage the decomposition from Proposition 1, estimating the expectation over the partial history h_τ with $N = 16$ realizations.

When evaluating fully amortized policies, we employ the sPCE (6) lower bound and sNMC (7) upper bound using a large number of contrastive samples, $L = 100000$, drawn from the prior $p(\theta)$ to approximate the inner expectation. The outer expectation is approximated using $N = 2048$ draws from the model $p(\theta)p(h_T \mid \theta, \pi)$. To approximate this quantity for Step-DAD efficiently, we use

$$\Delta\mathcal{I}(\pi^{s(\tau)}, \pi_{h_0}) \coloneqq \mathcal{I}_{1\to T}(\pi^{s(\tau)}) - \mathcal{I}_{1\to T}(\pi_{h_0}) \tag{10}$$

$$= \mathbb{E}_{p(h_\tau|\pi)}[\mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi^{s(\tau)}) - \mathcal{I}^{h_\tau}_{\tau+1\to T}(\pi_{h_0})] \tag{11}$$

$$\gtrsim \frac{1}{N}\left(\mathcal{L}^{h_\tau}_{\tau+1\to T}(\pi^{s(\tau)}) - \mathcal{U}^{h_\tau}_{\tau+1\to T}(\pi_{h_0})\right), \tag{12}$$

and add that difference to the lower bound estimate of $\mathcal{I}_{1\to\tau}(\pi)$.

D.3 Baselines

Static The Static (fixed) baseline pre-selects a fixed $\xi_1, ..., \xi_T$ ahead of the experiment. This non-adaptive approach used sPCE bound to optimize the design set $\xi_1, ..., \xi_T$.

Step-Static Step-Static computes a set of designs for $\xi_1, ..., \xi_\tau$ before a posterior update and subsequent computation of designs $\xi_\tau, ..., \xi_T$.

Random As the name implies, this baseline selects a random sample of designs $\xi_1, ..., \xi_T$. Thus the most non-informed naive approach.

D.4 Location Finding

The objective of the experiment is to ascertain the location, θ, of K sources. K is presumed to be predetermined. The intensity at each selected design choice, ξ, represents a noisy observation $\log y \mid \theta, \xi$ centered around the logarithm of the underlying model, $\mu(\theta, \xi)$.

$$\mu(\theta, \xi) = b + \sum_{k=1}^{K} \frac{\alpha_k}{(m + ||\theta_k - \xi||)^2} \tag{13}$$

In the given context, α_k may be either predetermined constants or random variables, $b > 0$ represents a fixed background signal, and m is a constant representing maximum signal.

$$\log[y \mid \theta, \xi] \sim \mathcal{N}(\log \mu(\theta, \xi), \sigma^2) \tag{14}$$

We assumed a normal standard prior at training: $\theta_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0_d, I_d)$

D.4.1 Training details

The model hyper parameters used are outlined in Table 5.

D.4.2 Optimal tuning step

The optimal value for EIG difference occurs at a range around $\tau = 6$ (Table 8). As expected, for small values of τ, small values for EIG difference are observed as there has not been a significant number of rollouts for the experiment to deviate from the pre-training data. Past

Table 5: **Source location finding.** Parameter Values

Parameter	Value
α_k	1 for all k
Max signal, m	10^{-4}
Base signal, b	10^{-1}
Observation noise scale, σ	0.5

Table 6: **Source location finding.** Parameters for training pre-training DAD/Step-DAD

Parameter	Value
Batch size	1024
Number of negative samples	1023
Number of gradient steps (default)	50000
Learning rate (LR)	0.0001

the peak at $\tau = 6$, the EIG difference once again drops as there are not enough experimental steps left to exploit any benefits which would arise from finetuning on the now extensive experimental history. The remaining decision space is too small.

D.4.3 Scaling up

As a further ablation, we test the robustness of a semi-amortized approach to the more complex task of location finding with multiple sources of signal. We find a positive EIG difference in all cases, once again demonstrating the benefits of using the semi-amortized Step-DAD network compared to the baseline fully amortized DAD network. Increasing the number of sources leads to a reduction in the EIG difference. However, this is expected given the increasing complexity of the task compared to the fixed number of steps post τ to adjust the decision making policy in the semi amortized setting. All experiments were run with $\tau = 7$.

Appendix E. Hyperbolic Temporal Discounting

Building on Foster et al. (2021), Mazur (1987) and Vincent (2016), we consider a hyperbolic temporal discounting model. A participant's behaviour is characterized by the latent variables $\theta = (k, \alpha)$ with prior distributions as follows:

$$\log k \sim \mathcal{N}(-4.25, 1.5) \qquad \alpha \sim \text{HalfNormal}(0, 2) \tag{15}$$

HalfNormal distribution denotes a Normal distribution truncated at 0. For given k, α, the value of the two propositions "£R today" and "£100 in D days" with design $\xi = (R, D)$ are given by:

$$V_0 = R, \qquad V_1 = \frac{100}{1 + kD} \tag{16}$$

Table 7: **Source location finding.** Parameters for Step-DAD finetuning

Parameter	Value
Num of theta rollouts	16
Number of posterior samples	20000
Finetuning learning rate (LR)	0.0001

Table 8: **Source location finding.** Total EIG for Step-DAD for various tuning steps τ.

τ	**Lower bound**	**Upper bound**
1 (Worst)	**6.826 (± 0.065)**	**6.837 (± 0.065)**
2	7.187 (± 0.097)	7.196 (± 0.098)
3	7.403 (± 0.098)	7.409 (± 0.099)
4	7.588 (± 0.103)	7.592 (± 0.104)
5	7.536 (± 0.183)	7.540 (± 0.183)
6 (Best)	**7.605 (± 0.078)**	**7.609 (± 0.078)**
7	7.560 (± 0.149)	7.568 (± 0.149)
8	7.382 (± 0.123)	7.395 (± 0.123)
9	7.255 (± 0.069)	7.277 (± 0.069)
DAD	6.771 (± 0.012)	6.803 (± 0.013)

Participants select V_1 in place of V_0 with probability modelled as:

$$p(y = 1|k, \alpha, R, D) = \epsilon + (1 - 2\epsilon)\Phi\left(\frac{V_1 - V_0}{\alpha}\right) \tag{17}$$

We fix $\epsilon = 0.01$ and ϕ is the c.d.f of the standard Normal Distribution.

$$\Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp -\frac{1}{2}z^2 \tag{18}$$

As in Foster et al. (2021), the design parameters R, D have the constraints $D > 0$ and $0 < R < 100$. R, D are represented in an unconstrained space ξ_d, ξ_r and transformed using the below maps.

$$D = \exp(\xi_d) \qquad R = 100 \cdot \text{sigmoid}(\xi_r) \tag{19}$$

Tables 9 and 10 give the hyperparameters for training the DAD/Step-DAD policies for the hyperbolic temporal discounting model

Table 9: **Hyperbolic Temporal Discounting model.** Parameters for training pre-training DAD/Step-DAD

Parameter	Value
Batch size	1024
Number of negative samples	1023
Number of gradient steps (default)	100000
Learning rate (LR)	5×10^{-5}
Annealing frequency	1000
Annealing factor	0.95
StepDAD: number of posterior draws	20000

Table 10: **Hyperbolic Temporal Discounting model.** Parameters for Step-DAD fine-tuning

Parameter	Value
Num of theta rollouts	16
Number of posterior samples	20000
Finetuning learning rate (LR)	5×10^{-5}

References

Anthony Atkinson, Alexander Donev, and Randall Tobias. *Optimum Experimental Designs, with SAS*. Oxford University Press, 2007. 1

Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017. 8

Warren K Bickel, Roberta Freitas-Lemos, Devin C Tomlinson, William H Craft, Diana R Keith, Liqa N Athamneh, Julia C Basso, and Leonard H Epstein. Temporal discounting as a candidate behavioral marker of obesity. *Neuroscience & Biobehavioral Reviews*, 129: 307–329, 2021. 10

Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 2018. 13

Tom Blau, Edwin Bonilla, Amir Dezfouli, and Iadine Chadès. Optimizing sequential experimental design with deep reinforcement learning. *arXiv preprint arXiv:2202.00821*, 2022. 1, 4, 7

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995. 1

Thomas S Critchfield and Scott H Kollins. Temporal discounting: Basic research and the analysis of socially important behavior. *Journal of applied behavior analysis*, 34(1): 101–122, 2001. 10

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006. 8

Christopher C Drovandi, James M McGree, and Anthony N Pettitt. A sequential monte carlo algorithm to incorporate model uncertainty in bayesian sequential design. *Journal of Computational and Graphical Statistics*, 23(1):3–24, 2014. 8

Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. *arXiv preprint arXiv:2101.11665*, 2021. 8

Adam Foster, Martin Jankowiak, Elias Bingham, Paul Horsfall, Yee Whye Teh, Thomas Rainforth, and Noah Goodman. Variational Bayesian Optimal Experimental Design. In *Advances in Neural Information Processing Systems 32*, pages 14036–14047. Curran Associates, Inc., 2019. 3, 6, 7

Adam Foster, Martin Jankowiak, Matthew O'Meara, Yee Whye Teh, and Tom Rainforth. A unified stochastic gradient approach to designing bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pages 2959–2969. PMLR, 2020. 3, 7

Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential bayesian experimental design. *Proceedings of the 38th International Conference on Machine Learning (ICML), PMLR 139*, 2021. 1, 3, 6, 7, 8, 10, 12, 13, 15, 16

Adam Evan Foster. *Variational, Monte Carlo and Policy-Based Approaches to Bayesian Experimental Design*. PhD thesis, University of Oxford, 2021. 1

Charles CJ Frye, Ann Galizio, Jonathan E Friedel, W Brady DeHart, and Amy L Odum. Measuring delay discounting in humans using an adjusting amount task. *JoVE (Journal of Visualized Experiments)*, (107):e53584, 2016. 10

Jinwoo Go and Tobin Isaac. Robust expected information gain for optimal bayesian experimental design using ambiguity sets. In *Uncertainty in Artificial Intelligence*, pages 728–737. PMLR, 2022. 4, 8

Daniel D Holt, Leonard Green, and Joel Myerson. Is discounting impulsive?: Evidence from temporal and probability discounting in gambling and non-gambling college students. *Behavioural processes*, 64(3):355–367, 2003. 10

Xun Huan and Youssef Marzouk. Gradient-based stochastic optimization methods in bayesian experimental design. *International Journal for Uncertainty Quantification*, 4(6), 2014. 7

Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013. 8

Xun Huan and Youssef M Marzouk. Sequential bayesian optimal experimental design via approximate dynamic programming. *arXiv preprint arXiv:1604.08320*, 2016. 1, 7

Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael Gutmann, and Tom Rainforth. Implicit Deep Adaptive Design: Policy–Based Experimental Design without Likelihoods. In *Advances in Neural Information Processing Systems*, volume 34, pages 25785–25798. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper/2021/file/d811406316b669ad3d370d78b51b1d2e-Paper.pdf`. 1, 4, 6, 7

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

Kris N Kirby. One-year temporal stability of delay-discount rates. *Psychonomic bulletin & review*, 16(3):457–462, 2009. 10

S. Kleinegesse and M.U. Gutmann. Efficient Bayesian experimental design for implicit models. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pages 1584–1592. PMLR, 2019. 7

Steven Kleinegesse and Michael Gutmann. Bayesian experimental design for implicit models by mutual information neural estimation. In *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 5316–5326. PMLR, 2020. 3, 6, 7

Steven Kleinegesse and Michael U. Gutmann. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds. *arXiv preprint arXiv:2105.04379*, 2021. 7

Steven Kleinegesse, Christopher Drovandi, and Michael U. Gutmann. Sequential Bayesian Experimental Design for Implicit Models via Mutual Information. *Bayesian Analysis*, pages 1 – 30, 2021. doi: 10.1214/20-BA1225. 7

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR, 2019. 7

Vincent Lim, Ellen Novoseller, Jeffrey Ichnowski, Huang Huang, and Ken Goldberg. Policy-based bayesian experimental design for non-differentiable implicit models. *arXiv preprint arXiv:2203.04272*, 2022. 1, 4, 7

Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956. 1, 2

Dennis V Lindley. *Bayesian statistics, a review*, volume 2. SIAM, 1972. 1

J. Lintusaari, M.U. Gutmann, R. Dutta, S. Kaski, and J. Corander. Fundamentals and recent developments in approximate Bayesian computation. *Systematic Biology*, 66(1): e66–e82, January 2017. 8

David JC MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992. 1

James E Mazur. An adjusting procedure for studying delayed reinforcement. *Commons, ML.; Mazur, JE.; Nevin, JA*, pages 55–73, 1987. 10, 15

Jessica McClelland, Bethan Dalton, Maria Kekic, Savani Bartholdy, Iain C Campbell, and Ulrike Schmidt. A systematic review of temporal discounting in eating disorders and obesity: Behavioural and neuroimaging findings. *Neuroscience & Biobehavioral Reviews*, 71:506–528, 2016. 10

Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020. 7

Jay I Myung, Daniel R Cavagnaro, and Mark A Pitt. A tutorial on adaptive design optimization. *Journal of mathematical psychology*, 57(3-4):53–67, 2013. 1, 7

Xuanlong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11), 2010. ISSN 00189448. doi: 10.1109/TIT.2010.2068870. 6, 13

Antony Overstall and James McGree. Bayesian Design of Experiments for Intractable Likelihood Models Using Coupled Auxiliary Models and Multivariate Emulation. *Bayesian Analysis*, 15(1):103 – 131, 2020. doi: 10.1214/19-BA1144. 7

Antony Overstall and James McGree. Bayesian decision-theoretic design of experiments under an alternative model. *Bayesian Analysis*, 17(4):1021–1041, 2022. 4, 8

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 13

David J Price, Nigel G Bean, Joshua V Ross, and Jonathan Tuke. An induced natural selection heuristic for finding optimal bayesian experimental designs. *Computational Statistics & Data Analysis*, 126:112–124, 2018. 7

Tom Rainforth. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, University of Oxford, 2017. 8

Tom Rainforth, Rob Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pages 4267–4276. PMLR, 2018. 3, 6

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern bayesian experimental design. *arXiv preprint arXiv:2302.14545*, 2023. 1

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pages 1278–1286, 2014. 7

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951. 7

Elizabeth G Ryan, Christopher C Drovandi, James M McGree, and Anthony N Pettitt. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016. 1, 3

Wanggang Shen and Xun Huan. Bayesian sequential optimal experimental design for nonlinear models using policy gradient reinforcement learning. *CoRR*, abs/2110.15335, 2021. URL `https://arxiv.org/abs/2110.15335`. 3

Xiaohong Sheng and Yu Hen Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 2005. ISSN 1053587X. doi: 10.1109/TSP.2004.838930. 8

S.A. Sisson, Y. Fan, and M. Beaumont. *Handbook of Approximate Bayesian Computation*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, 2018. ISBN 9781351643467. 8

Giles W Story, Ivo Vlaev, Ben Seymour, Ara Darzi, and Raymond J Dolan. Does temporal discounting explain unhealthy behavior? a systematic review and reinforcement learning perspective. *Frontiers in behavioral neuroscience*, 8:76, 2014. 10

Linda M Tate, Pao-Feng Tsai, Reid D Landes, Mallikarjuna Rettiganti, and Leanne L Lefler. Temporal discounting rates and their relation to exercise behavior in older adults. *Physiology & behavior*, 152:295–299, 2015. 10

Owen Thomas, Ritabrata Dutta, Jukka Corander, Samuel Kaski, and Michael U Gutmann. Likelihood-free inference by ratio estimation. *arXiv preprint arXiv:1611.10242*, 2016. 8

Raman Uppal and Tan Wang. Model misspecification and underdiversification. *The Journal of Finance*, 58(6):2465–2486, 2003. 4

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 6, 13

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 7

Benjamin T Vincent. Hierarchical bayesian estimation and hypothesis testing for delay discounting tasks. *Behavior research methods*, 48(4):1608–1620, 2016. 10, 15

Benjamin T Vincent and Tom Rainforth. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *PsyArXiv preprint*, 2017. 7, 10

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 7

M. Zaharia, Andrew Chen, A. Davidson, A. Ghodsi, S. Hong, A. Konwinski, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, Fen Xie, and Corey Zumar. Accelerating the machine learning lifecycle with MLflow. *IEEE Data Eng. Bull.*, 41:39–45, 2018. 13

Manzil Zaheer, Satwik Kottur, Siamak Ravanbhakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J Smola. Deep sets. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 2017. 7

Cheng Zhang, Judith Bütepage, Hedvig Kjellstrom, and Stephan Mandt. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 41 (8):2008–2026, 2018. 8