



DEPARTMENT OF
STATISTICS

Automated Data Acquisition via Bayesian Experimental Design

A thesis submitted for the degree of
Doctor of Philosophy

MICHAELMAS 2024

DESI R. IVANOVA

DEPARTMENT OF STATISTICS

ST PETER'S COLLEGE

UNIVERSITY OF OXFORD

Abstract

Acquiring high-quality data is a major challenge in science and engineering. Data collection, whether through large-scale online surveys or carefully conducted laboratory experiments, is often costly, time-consuming, and constrained by limited resources. Consequently, designing optimal experiments to gather informative and valuable data is crucial for the efficient allocation of resources and, ultimately, for making better decisions.

Bayesian experimental design (BED) provides a principled mathematical framework for designing experiments to efficiently learn about a phenomenon of interest. This thesis focuses on information-theoretic BED, employing expected information gain (EIG) as the design criterion. Maximising the EIG ensures that the collected data is most informative about the underlying scientific question or hypothesis. This approach has the potential to significantly enhance the quality and efficiency of data acquisition, addressing the critical challenges of high costs, time constraints, and limited resources.

Despite its potential, the practical application of BED has historically been limited by significant computational challenges. The primary challenge lies in calculating the EIG, which is intractable for most realistic problems. Furthermore, adaptive experimental design, which leverages the information gained from previous experiments to guide the design of subsequent ones, adds an additional layer of computational complexity.

This thesis focuses on developing scalable and computationally efficient BED approaches to overcome these challenges. We introduce a novel class of methods called policy-based BED (PB-BED), which leverages deep learning to fully amortise the cost of adaptive experimental design, enabling real-time design decisions. Additionally, we introduce a semi-amortised approach that allows for periodic refinement of the design policy during the experiment itself, based on the actual data collected. This approach enhances the adaptability and robustness of the PB-BED framework, ensuring that the design strategy remains optimal as new information is gathered. Finally, we introduce a unified model-agnostic framework for designing large-scale contextual experiments using information-theoretic principles.

The contributions made in this thesis represent a step towards automated and reliable experimental design strategies that have the potential to accelerate scientific discovery and improve data-driven decision making across various domains.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Tom Rainforth and Yee Whye Teh, for their unwavering support, guidance, and encouragement throughout my DPhil. Thank you, Tom, for your constant support and your invaluable advice on how to conduct, write, and present research. Your mentorship has been crucial to my success during my DPhil and to my future academic endeavours. I also appreciate your support through my internship spree. To Yee Whye, thank you for your support, guidance, and the insightful discussions during the DeepProb reading groups. Your mentorship has expanded my research interests and helped me become a more well-rounded researcher by exposing me to a wide range of fields.

I am immensely thankful to my collaborators and co-authors. Special thanks to Adam Foster for his guidance and support from the first days of my DPhil up until now; I feel incredibly fortunate to have had you as my first collaborator and to have learned so much from you. I extend my gratitude to Michael Gutmann, Marvin Schmitt, Paul-Christian Bürkner and Stefan Radev for the continued collaborations, as well as Freddie Bickford Smith, Steven Kleinegesse, Marcel Hedman, Panos Tigas, Yashas Annadani, Andrew Jesson, and Stefan Bauer.

I would like to thank my internships supervisors and mentors. At Microsoft, I thank Cheng Zhang and Adam Foster for their technical guidance, Joel Jennings for his help with engineering, as well as the wider Causica team for their support. Thank you to Karen Ullrich and Mark Tygert for the incredible opportunity to intern at FAIR (Meta AI) in NYC. Special thanks to Matthew Muckley for his mentorship and support during the internship, both in engineering and research. Interning at FAIR was truly an exceptional experience; it provided a unique opportunity to meet, speak to, and learn from so many brilliant researchers, which profoundly impacted my views on ML and AI research. I also want to thank fellow Meta interns and friends, including Andreea, Karan, Julius, Sagar, Winnie and Halil, to name a few, for making my time in NYC unforgettable. And thank you, NYC, for making me appreciate Europe in a different kind of way.

To my fellow RainML lab members and the wider OxCSML group—thank you for creating

a supportive and intellectually stimulating environment. Our discussions, friendships and the occasional board game nights have made my DPhil journey more enjoyable and rewarding.

I would like to thank the Statistics Department for their support and the StatML CDT for providing financial support. I am also grateful to my examiners throughout my DPhil—Michael Osborne, Arnaud Doucet, and Seth Flaxman—for their valuable feedback. Special thanks to Michael Osborne and Samuel Kaski, who will be assessing this thesis, for their time and expertise.

Last but not least, I thank my partner Momchil for inspiring me to pursue the DPhil in the first place, and for his patience, support and encouragement throughout it. Thank you for being my biggest supporter, I could not have done this without you.

Contents

1	Preliminaries	7
1.1	Introduction	7
1.1.1	Brief history of optimal experimental design	8
1.1.2	Research contributions and thesis outline	9
1.2	Background	11
1.2.1	The Bayesian approach to learning	11
1.2.2	Information theory basics	16
1.2.3	Bayesian experimental design (BED)	21
1.3	Literature review: Computational methods for BED	27
1.3.1	EIG estimation	27
1.3.2	EIG optimisation	37
1.3.3	Multiple experiments	41
1.4	Beyond parameter learning: Goal-oriented Bayesian experimental design	47
1.4.1	EIG formulation and estimation for goal-oriented BED	49
1.5	Related fields: Bayesian optimisation, Bayesian active learning and Bayesian Reinforcement Learning	52
1.5.1	Bayesian optimisation	52
1.5.2	Bayesian active learning	53
1.5.3	Bayesian model-based reinforcement learning	54
	Chapter references	57
2	Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design	72

Manuscript	73
Statement of authorship	101
3 Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods	102
Manuscript	103
Statement of authorship	136
4 Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design	137
Manuscript	138
Statement of authorship	153
5 CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design	154
Manuscript	155
Statement of authorship	175
6 Discussion	176
6.1 Frequently asked questions (FAQs)	176
6.2 Limitations and future work	179
6.2.1 Model misspecification	179
6.2.2 Evaluating BED methods	181
6.2.3 PB-BED, model-based Reinforcement Learning and Active Inference . . .	182
6.2.4 Engineering	183
6.2.5 Real-world applications	184
6.3 Conclusion	186
Chapter references	187
7 Bibliography	191
List of Publications	191
Combined Bibliography	192

Chapter 1

Preliminaries

1.1 Introduction

Data plays a pivotal role in the modern world, serving as the foundation for both scientific discoveries and industry advancements. In recent years, sophisticated statistical models trained on large datasets have been increasingly employed to guide decision-making processes in an automated and autonomous manner (Araujo et al., 2020; Bogert et al., 2021; Adlung et al., 2021). However, the quality and relevance of the data used to train these models can vary significantly. Datasets may be affected by noise, redundancy, or bias, or may not be directly applicable to the specific tasks at hand. When models are trained on suboptimal data, they tend to inherit and even exacerbate the biases and limitations present in it. The resulting models may then produce biased analyses and predictions, which can lead to misleading conclusions and poor downstream decision-making (Barocas and Selbst, 2016; Corbett-Davies et al., 2017; Ntoutsis et al., 2020; Paullada et al., 2021; Mehrabi et al., 2021; Budach et al., 2022). Consequently, high-quality and relevant data is an invaluable asset in the modern data-driven landscape (Hutchinson et al., 2021; Paleyes et al., 2022; Priestley et al., 2023).

In some cases, data is entirely generated by external sources, leaving no control over the data acquisition process and necessitating the use of available data as-is. However, in many situations, opportunities exist to influence the data acquisition process, potentially leading to improvements in the quality of the collected data. These opportunities include carefully choosing survey questions, selecting promising compounds for drug discovery trials, calibrating laboratory equipment for precise measurements, and strategically choosing data points for annotation in machine learning tasks. Such controllable variables offer crucial opportunities to maximise the usefulness and quality of the acquired data.

This controlled data acquisition process can be formalised through the lens of *optimal experimental design* (Fisher, 1935; Ryan and Morgan, 2007). In this work, we adopt the Bayesian approach to designing experiments (Lindley, 1956; Chaloner and Verdinelli, 1995; Ryan et al., 2016; Rainforth et al., 2024), as it provides a unified, coherent, and self-consistent framework. Bayesian experimental design (BED) is a model-based framework that effectively integrates all available information, making it particularly advantageous in multi-step *adaptive* experiment settings, where decisions are made sequentially with incomplete information. This allows for consistent updates to the model, ensuring that each decision aligns with the evolving understanding of the experimental system.

1.1.1 Brief history of optimal experimental design

Optimal experimental design (OED) emerged in the early 20th century within the framework of classical, frequentist statistics. The origin of the field is commonly attributed to Sir Ronald A. Fisher, whose 1935 book, *The Design of Experiments*, formalised foundational techniques such as analysis-of-variance, factorial design, confounding, and provided applications of data collection and analysis (Fisher, 1935; Box, 1980). However, the conceptual foundations of OED can be traced back further to the doctoral work of Danish statistician Kirstine Smith. In her thesis, Smith (1918) discusses how the residual error variance could guide the design of experiments where the resulting data are used to fit polynomial linear models from first to sixth order. This work introduced what we now refer to as *G*-optimal designs for single-factor experiments. During this early period, a significant focus was on parameter learning for *linear models*, leading to the development of several “alphabetic” optimality criteria. Examples of these include *A*-, *D*-, *E*-, and *G*-optimality (Smith, 1918; Box, 1982), *c*-optimality (Elfving, 1952), and *T*-optimality (Atkinson and Fedorov, 1975). These criteria, based on the Fisher information matrix, are computationally tractable and well-suited for the predominantly used linear models at the time. However, they yield only locally optimal designs in non-linear settings (Ryan et al., 2016).

The Bayesian approach to optimal experimental design has its roots in information theory (Shannon, 1948). Specifically, Dennis V. Lindley mathematically formalised the notion of information provided by an experiment, establishing the concept of *expected information gain* (EIG) as a principled criterion for choosing designs (Lindley, 1956). The EIG quantifies the anticipated reduction in uncertainty about the model parameters after observing the experimental outcomes, as measured by the expected reduction in Shannon entropy between the prior and posterior distributions on those parameters. Drawing on principles from decision theory (Raiffa and Schlaifer, 1961), Lindley later formulated a more general, decision-theoretic approach to design-

ing optimal experiments, utilising an arbitrary utility function as a design criterion (Lindley, 1972). Under this broader utility-based framework, it can be shown that some of the frequentist alphabetic optimality criteria have corresponding Bayesian analogs (Chaloner and Verdinelli, 1995; Ryan et al., 2016). The Bayesian approach to experimental design, however, offers several advantages over classical methods, including the ability to handle non-linear models, incorporate prior knowledge, and integrate uncertainty and risks into the decision-making process.

The increase in computational resources and the development of simulation-based methods have led to the emergence of many model-specific Bayesian optimal design methodologies in the literature. For example, Han and Chaloner (2004) consider nonlinear mixed-effects models, Cook et al. (2008) focus on epidemiology models, and Müller et al. (2006) examine dose-finding. A comprehensive review of the developments in Bayesian experimental design during this period can be found in Ryan et al. (2016).

Recent advancements in scalable variational methods, employing neural networks to approximate unknown densities, have significantly enhanced the capabilities and practical applicability of BED methods. These improvements facilitate more efficient estimation and optimisation of information objectives in a model-agnostic manner (Foster et al., 2019, 2020; Kleinegesse and Gutmann, 2020, 2021; Go and Isaac, 2022). However, these approaches are designed for a single experiment or multiple non-adaptive experiments. The challenge of performing multiple experiments adaptively has remained somewhat under-explored. This limitation restricts the applicability of BED, particularly in scenarios requiring real-time decision-making where adaptive strategies could provide significant advantages. The computational demands of adaptive BED, which involves iterative updates and optimisations based on newly acquired data, make it difficult to deploy in settings where quick data processing and decision-making are crucial.

1.1.2 Research contributions and thesis outline

Recent advancements in non-adaptive BED, which include static and single-step methods, have been substantial. A variety of scalable and robust techniques are now available for optimising single experiments (Foster et al., 2019; Kleinegesse and Gutmann, 2020; Go and Isaac, 2022). However, adaptive BED, which allows for the incorporation of past data or other forms of personalisation into design decisions, presents distinct challenges. Traditional methods in adaptive BED often fall short in terms of scalability and real-time processing capabilities, especially when applied to complex, high-dimensional models. These methods tend to be very computationally intensive, thereby hindering their practical application in scenarios that demand quick decision-

Mathematical Formulation	Computational Methods	BED + Deep Learning
• decision-theoretic approach • parametric linear models • <i>for a review, see</i> Chaloner and Verdinelli (1995)	• many are model-specific • low-dimensional problems • <i>for a review, see</i> Ryan et al. (2016)	• model-agnostic • scalable in high-dimensions • <i>for a review, see</i> Rainforth et al. (2024)
1950–2000	2000–2020	2020–present



Figure 1.1: **Timeline of key developments in Bayesian Experimental Design (BED).** The field was pioneered by Lindley (1956), shortly following the foundational work of Shannon in information theory (Shannon, 1948). The 2000s and 2010s saw the development of numerous model-specific computational methods. Most recently, those methods have evolved into model-agnostic, scalable approaches powered by flexible deep learning modules, significantly expanding effectiveness and scope.

making. Addressing these limitations is crucial for the effective implementation of adaptive BED in various domains.

This thesis focuses on advancing the field of adaptive BED by addressing the critical requirements for automated, real-time updates in a scalable and model-agnostic fashion. The key innovations introduced in this thesis are as follows:

- A novel, fully amortised, policy-based adaptive BED method, called **Deep Adaptive Design (DAD)**, that leverages deep learning to eliminate the computational overhead associated with the sequential design problem, facilitating real-time design decisions. This method is introduced in Chapter §2.
- An extension of DAD to support models without explicit likelihood functions (**implicit Deep Adaptive Design, or iDAD**). Similar to DAD, iDAD is fully amortised, and offers real-time experimental design for differentiable simulator models. This method is introduced in Chapter §3.
- A semi-amortised policy-based BED approach that enables adaptivity of the policy itself, resulting in a more robust yet still computationally efficient method for adaptive experimentation. This method is introduced in Chapter §4.
- An information-theoretic approach to **Contextual Optimisation via BED (CO-BED)**. Here experiments are performed in parallel (i.e. not sequentially), and adaptation of the experimental designs is achieved by conditioning on instance-specific contextual information. This method is introduced in Chapter §5.

Chapter §6 identifies key challenges and opportunities in the field of BED and discusses potential directions for future research.

The remainder of this chapter is structured as follows: Section §1.2 introduces the technical background necessary for understanding the content presented throughout this thesis, while Section §1.3 provides a comprehensive review of the related literature on previous BED approaches, highlighting their strengths and limitations.

1.2 Background

This section outlines the foundational concepts critical to the methodologies developed in this thesis. It begins with an overview of the Bayesian approach to learning, explaining how Bayesian principles facilitate the inference of unknown parameters from data (§1.2.1). Next, §1.2.2 introduces the core concepts from information theory, which form the basis of Bayesian experimental design (BED). Finally, §1.2.3 introduces the main principles of BED and its key objective.

1.2.1 The Bayesian approach to learning

Statistical models are essential tools for describing and understanding real-world phenomena. These models typically express an observed quantity $y \in \mathcal{Y}$ as a stochastic function of some unknown parameters, sometimes called latent variables, $\theta \in \Theta$, and some covariates $x \in \mathcal{X}$, which might be fixed (i.e. predefined, including $x = \emptyset$), stochastic (and modelled independently), or directly controlled (i.e. designed). The main goal is to *infer*, *fit* or *learn* the values of the parameters θ that best explain some observed data (x, y) .

The Bayesian framework offers a principled and coherent approach to statistical modelling (Bernardo and Smith, 2009; Gelman et al., 2013). It treats the parameters θ as random variables with associated probability distributions that reflect the inherent uncertainty about their true values. Initial beliefs and knowledge about the plausible values for θ are encoded through the **prior distribution** $p(\theta)$, which can be informed by domain expertise, previous empirical studies, or theoretical considerations. Priors are usually specified *explicitly* in closed form, allowing for direct evaluation of the prior density $p(\theta)$. Occasionally, priors might only be defined *implicitly*, meaning the exact analytical form of $p(\theta)$ is not specified but we can draw samples from it. The process of specifying the prior distribution is known as *prior elicitation* and constitutes a crucial step in the Bayesian workflow (Winkler, 1967; O’Hagan, 2019; Gelman et al., 2020; Mikkola et al., 2021).

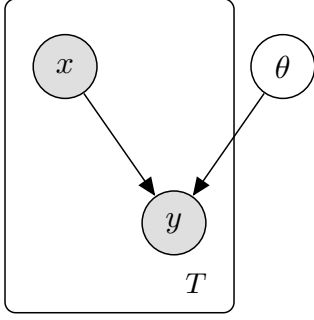


Figure 1.2: Graphical representation of the Bayesian model we consider. The goal is to infer the unobserved parameters θ from a dataset $h_T = \{x_i, y_i\}_{i=1}^T$ containing T observed covariate-outcome pairs. This dataset can be available in its entirety from the outset (offline or batch learning) or arrive sequentially one observation at a time (online learning).

The relationship between the observed data y with covariates x , and the parameters θ is described by the **likelihood function** $p(y|\theta, x)$. As with the prior, the likelihood can be *explicit*, with an analytic closed-form expression, allowing direct computation of $p(y|\theta, x)$ —the probability of observing y given θ and x , or *implicit*, meaning that we can sample $y|\theta, x$ but cannot evaluate its density. Implicit models are common in real-world applications as they make fewer assumptions about the underlying data-generating process; however, as we will see later, this flexibility trades-off against computational efficiency.

Remark 1.1. *We will use the notation $p(\theta)$ and $p(y|\theta, x)$, to denote the prior and likelihood functions, respectively, regardless of whether they are implicit or explicit. In cases where this distinction is important, we will highlight the relevant assumptions and discuss any implications for the methods and results presented.*

The prior distribution and the likelihood function together form a joint generative model for the underlying data generating process (DGP) of interest $p(\theta, y|\xi) = p(\theta)p(y|\theta, x)$. Given an observed value y associated with a covariate x , we can update the prior to a *posterior* distribution by applying Bayes’ rule

$$p(\theta|x, y) = \frac{p(\theta, y|x)}{p(y|x)}. \quad (1.1)$$

The denominator, known as the *Bayesian marginal likelihood*, *model evidence*, or *prior predictive distribution*, normalises the posterior probability and is constant with respect to θ . Computing it involves integrating over all possible values for θ

$$p(y|x) = \int_{\theta \in \Theta} p(\theta)p(y|\theta, x) d\theta = \mathbb{E}_{p(\theta)} [p(y|\theta, x)]. \quad (1.2)$$

Computing this marginal exactly is often intractable, as the integral required is not available in closed form. Analytic solutions exist only in special cases, such as conjugate models, where the prior and likelihood distributions are chosen to ensure mathematical convenience (Diaconis and Ylvisaker, 1979). As a result, the posterior is in general intractable and we resort to computational methods to approximate it. We briefly discuss the main approaches next.

Computational methods for posterior inference

The core challenge in the Bayesian approach to learning is evaluating the posterior distribution (1.1), which is usually intractable due to the intractable marginal likelihood (1.2). There are two main computational strategies that aim to address this intractability: sample-based approaches and optimisation-based approaches.

Sample-based approaches, such as Markov Chain Monte Carlo (MCMC), are the foundational technique for Bayesian inference. These methods generate samples from the posterior distribution by constructing a Markov chain whose equilibrium distribution matches that of the posterior. Prominent algorithms include the Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970), the Gibbs sampler (Geman and Geman, 1984), and Hamiltonian Monte Carlo (HMC, see Betancourt, 2017, for a review), among others.

Optimisation-based approaches, such as Variational Inference (VI), aim to find the best approximation of the posterior within a constrained family of distributions (Wainwright et al., 2008; Blei et al., 2017; Zhang et al., 2018). These methods reformulate Bayesian inference as an optimisation problem, where the objective is to minimise a measure of dissimilarity between the true posterior and its approximation.

Practical considerations. The choice between sample-based and optimisation-based approaches depends on several technical and methodological considerations. MCMC methods provide asymptotically exact samples from the posterior distribution, but they can be computationally expensive and slow to converge, particularly for high-dimensional problems. Consequently, they may not be well-suited for large-scale applications or situations requiring rapid approximations. Moreover, sample-based approaches generally rely on an explicit likelihood function, rendering them inapplicable to models with intractable likelihoods. Lastly, sample-based methods, as their name suggests, do not produce closed-form posterior distributions, potentially restricting the scope of downstream applications for which they can be employed.

In contrast, optimisation-based approaches like VI produce a closed-form, explicit posterior within a chosen family of distributions. VI can handle both explicit and implicit likelihood functions. This approach is often faster than sample-based methods, making it appealing for large-scale problems or when quick approximations are needed. However, the speed comes at the cost of potentially introducing bias if the chosen family of distributions does not contain the true posterior. The statistical properties of VI are less well-understood, adding a layer of uncertainty to its reliability.

Offline vs online learning

In the context of statistical learning, including the Bayesian framework described here, we can distinguish between two main paradigms: offline (batch) learning and online learning.

In offline learning, we assume access to a fixed dataset $h_T = \{(x_i, y_i)\}_{i=1}^T$, consisting of T covariate-observation pairs. The goal is to learn the parameters θ by leveraging the entire dataset at once. In the Bayesian setting, this involves computing the posterior distribution $p(\theta | h_T)$. Offline learning is suitable when all the data is available simultaneously, and the model and inference procedure are not expected to change over time.

In contrast, online learning considers a setting where data arrives sequentially, one observation at a time. At time t , given access to the previous data $h_{t-1} = \{(x_i, y_i)\}_{i=1}^{t-1}$ and a new covariate-observation pair (x_t, y_t) , the objective is to update the parameters θ based on this new information. In the Bayesian framework, this involves computing the updated posterior distribution $p(\theta | h_t) \propto p(\theta | h_{t-1})p(y_t | \theta, x_t)$. Online learning allows for adaptivity and refinement of the model as new data becomes available, making it useful in scenarios with streaming data, evolving environments, or large datasets that cannot be processed in batch mode. However, it poses significant computational challenges due to the need for repeated posterior updates after each new observation, which, as discussed, is usually very computationally costly.

Illustrative examples

We present several illustrative examples to demonstrate various scenarios, including simple conjugate models with tractable posteriors, models with explicit likelihoods but intractable posteriors, and models with both intractable likelihoods and intractable posteriors.

We start with an example of a simple conjugate model with a tractable posterior.

Example 1.2 (Bayesian linear regression). *We model the relationship between outcomes $y \in \mathbb{R}$ and covariates $x \in \mathbb{R}^d$ as $y = \theta^\top x + \epsilon$, where $\theta \in \mathbb{R}^d$ represents the regression coefficients, and ϵ is the noise term, assumed to be normally distributed with known variance σ_ϵ^2 , $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$.*

- **Prior:** *The coefficients come from a multivariate normal distribution with zero mean and known covariance matrix, $\theta \sim \mathcal{N}(\mathbf{0}, \Sigma_\theta^{\text{prior}})$.*
- **Likelihood:** *Since θ and ϵ are both Gaussian, then we also have $p(y | x, \theta) = \mathcal{N}(y; \theta^\top x, \sigma_\epsilon^2)$.*

For a dataset of size N , $\{(x_i, y_i)\}_{i=1}^N$, we have

$$p(y_{1:N} | \theta, x_{1:N}) = \prod_{i=1}^N p(y_i | x_i, \theta) = \prod_{i=1}^N \mathcal{N}(y_i; \theta^\top x_i, \sigma_\epsilon^2) \quad (1.3)$$

- **Posterior:** Due to the conjugacy between the Gaussian prior and likelihood, the posterior distribution is also Gaussian and can be computed in closed form:

$$p(\theta | x_{1:N}, y_{1:N}) = \mathcal{N}(\theta; \mu_\theta^{post}, \Sigma_\theta^{post}), \quad (1.4)$$

where $\mu_\theta^{post} = \sigma_\epsilon^{-2} \Sigma_\theta^{post} X^\top y$, and $\Sigma_\theta^{post} = (\sigma_\epsilon^{-2} X^\top X + (\Sigma_\theta^{prior})^{-1})^{-1}$.

Next, we explore a scenario where both the prior and the likelihood are simple and explicit, but the resulting posterior complex and intractable.

Example 1.3 (Hidden object location finding). Suppose $\theta \in \mathbb{R}^2$ corresponds to the unknown location of an object. We observe the log-intensity of a signal it emits at some location $x \in \mathbb{R}^2$.

- **Prior:** The location is drawn from a standard normal distribution: $\theta \sim \mathcal{N}(\mathbf{0}_2, \mathbf{I}_2)$.
- **Likelihood:** The signal log-intensity observed at $\xi \in \mathbb{R}^2$ decreases according to the inverse-square law: $p(\log y | \theta, x) = \mathcal{N}(\log y; \log \mu(\theta, x), \sigma^2)$, where $\mu(\theta, x) = \delta_1 + \frac{1}{\delta_2 + \|\theta_k - x\|^2}$, with $\delta_{1,2} > 0$ fixed.
- **Posterior:** Due to the nonlinear relationship between θ and x in the likelihood model, the posterior distribution $p(\theta | y, x)$ is no longer Gaussian and not available in closed form.

Finally, we present an example of a model with an intractable likelihood and posterior.

Example 1.4 (Stochastic SIR model). We consider the Susceptible-Infectious-Recovered (SIR) model (Cook et al., 2008), a widely-used epidemiological model based on a system of stochastic differential equations (SDEs). The SIR model describes the dynamics of disease spread within a fixed population of size N . It assumes that at any given time $\tau > 0$, the population is divided into three compartments: susceptible individuals $S(\tau)$, infected individuals $I(\tau)$, and recovered individuals $R(\tau)$. Susceptible individuals can become infected and transition to the infected compartment, while infected individuals can recover from the disease and move to the recovered compartment. Assuming a constant total population size N , we define the state population vector as $\mathbf{Y}(\tau) = (S(\tau), I(\tau))^\top$. For the purpose of modeling, we focus on the dynamics of the susceptible and infected populations, noting that the recovered population can be derived from the conservation of total population: $R(\tau) = N - S(\tau) - I(\tau)$.

- **Prior:** The model parameters β and γ have log-normal priors

$$\beta \sim \text{Log-}\mathcal{N}(0.50, 0.50^2) \quad \gamma \sim \text{Log-}\mathcal{N}(0.10, 0.50^2). \quad (1.5)$$

- **Likelihood:** The system of Itô SDEs that defines the stochastic SIR model is given by

$$d\mathbf{Y}(\tau) = \mathbf{f}(\mathbf{X}(\tau))d\tau + \mathbf{G}(\mathbf{Y}(\tau))d\mathbf{W}(\tau), \quad (1.6)$$

where $\mathbf{f}$ is a drift vector, $\mathbf{G}$ is a diffusion matrix and $\mathbf{W}(\tau)$ is a vector of independent Wiener processes. The drift vector and diffusion matrix are given by

$$\mathbf{f}(\mathbf{X}(\tau)) = \begin{pmatrix} -\beta \frac{S(\tau)I(\tau)}{N} \\ \beta \frac{S(\tau)I(\tau)}{N} - \gamma I(\tau) \end{pmatrix} \quad \text{and} \quad \mathbf{G}(\mathbf{X}(\tau)) = \begin{pmatrix} -\sqrt{\beta \frac{S(\tau)I(\tau)}{N}} & 0 \\ \sqrt{\beta \frac{S(\tau)I(\tau)}{N}} & -\sqrt{\gamma I(\tau)} \end{pmatrix}. \quad (1.7)$$

The likelihood $p(y | \theta, \tau)$ is not available in closed form, as the SDE system does not have an analytic solution.

- **Posterior:** The posterior distribution $p(\theta | y, \tau)$ is intractable due to the intractable likelihood.

1.2.2 Information theory basics

Information theory has played a pivotal role in shaping the development of modern machine learning, providing a rigorous mathematical framework for quantifying and manipulating information. Its principles and concepts have inspired the development of information-based methodologies across many fields, including (neural) data compression and data acquisition. Examples of the latter include Bayesian experimental design, Bayesian optimisation, and Bayesian active learning, where we can leverage information-theoretic measures to guide data acquisition.

This subsection aims to review the core concepts from information theory that are relevant to the work presented in this thesis. For a more comprehensive overview of the field, please refer to Cover and Thomas (1999) or MacKay (2003).

Remark 1.5. All definitions and results are applicable to both continuous and discrete random variables and are presented in a way that covers both.

Definition 1.6 (Entropy, cross-entropy, relative entropy/KL divergence). Let Z be a random variable with distribution $p(z)$.

- The (Shannon) entropy of Z is

$$H[Z] = H[p(z)] = \mathbb{E}_{p(z)}[-\log p(z)]. \quad (1.8)$$

- The cross-entropy between the distribution $p(z)$ and another distribution on z , $q(z)$, is

$$H[p(z) \parallel q(z)] = \mathbb{E}_{p(z)}[-\log q(z)]. \quad (1.9)$$

- The relative entropy or Kullback-Leibler divergence from $p(z)$ to $q(z)$ is

$$KL(p(z) \parallel q(z)) = \mathbb{E}_{p(z)} \left[\log \frac{p(z)}{q(z)} \right] = H[p(z) \parallel q(z)] - H[p(z)]. \quad (1.10)$$

Entropy, defined in Equation (1.8), quantifies the average level of uncertainty inherent in the outcomes of a random variable Z . A higher entropy value indicates greater unpredictability or uncertainty in these outcomes. In contrast to other statistical measures of uncertainty, such as variance, entropy is a scale-invariant measure, meaning it remains unaffected by changes in the scale of the random variable Z .

Cross-entropy, defined in Equation (1.9), measures the discrepancy between two probability distributions, typically the true distribution $p(z)$ and a model distribution $q(z)$. In most practical settings, the true distribution $p(z)$ is unknown, and the goal is to approximate it with the model distribution $q(z)$. The Kullback-Leibler (KL) divergence, or relative entropy, defined in Equation (1.10), extends this concept by specifically quantifying the inefficiency of using the model distribution $q(z)$ when the true distribution is $p(z)$. Minimising the KL divergence $KL(p(z) \parallel q(z))$ is equivalent to minimising the cross-entropy $H[p(z) \parallel q(z)]$, as the entropy term $H[p(z)]$ is independent of the model distribution $q(z)$.

Minimising the KL divergence, or equivalently, minimising the cross-entropy between the true and model distributions, is a common objective in many machine learning tasks. Gibbs' inequality, introduced in the following proposition, demonstrates that both the KL divergence and cross-entropy achieve their minimal values when the true distribution $p(z)$ and the model distribution $q(z)$ are identical, i.e., $p \equiv q$.

Proposition 1.7 (Gibbs' inequality). *The relative entropy, or Kullback-Leibler divergence, between any two probability distributions on z , $p(z)$ and $q(z)$ is non-negative, i.e.*

$$KL(p(z) \parallel q(z)) \geq 0, \quad (1.11)$$

with equality if and only if $p \equiv q$. The relative entropy equals the difference between cross-entropy and entropy, so the inequality can also be written as:

$$H[p(z)] \leq H[p(z) \| q(z)]. \quad (1.12)$$

Proof. By Jensen's inequality we have

$$\text{KL}(p(z) \| q(z)) = -\mathbb{E}_{p(z)} \left[\log \frac{q(z)}{p(z)} \right] \geq -\log \mathbb{E}_{p(z)} \left[\frac{q(z)}{p(z)} \right] = -\log \mathbb{E}_{q(z)}[1] = 0 \quad (1.13)$$

with equality holding if and only if $p \equiv q$. □

Remark 1.8 (Intuition from compression). *The concept of Kullback-Leibler (KL) divergence can be intuitively understood through the lens of data compression. In information theory, the entropy $H[p(z)]$ of a distribution $p(z)$ represents the optimal expected codelength for encoding messages drawn from that distribution. If we use a model $q(z)$ to encode data that is actually distributed according to $p(z)$, the expected codelength becomes the cross-entropy $H[p(z) \| q(z)]$.*

The KL divergence, $\text{KL}(p(z) \| q(z))$, measures the extra bits required per data point due to the inefficiency of using the model $q(z)$ instead of the true distribution $p(z)$. This additional cost in the average description length is the "penalty" for assuming the wrong model. Thus, $\text{KL}(p(z) \| q(z))$ quantifies the overhead or inefficiency introduced by this assumption, offering a measure of how suboptimal the model $q(z)$ is when used to describe or compress data that was generated by $p(z)$.

Definition 1.9 (Joint and conditional entropy). *Let X and Y be two random variables with a joint probability distribution $p(x, y)$.*

- *The joint entropy $H[X, Y]$ is defined as*

$$H[X, Y] = \mathbb{E}_{p(x, y)}[-\log p(x, y)]. \quad (1.14)$$

- *The conditional entropy $H[X | Y]$ is defined as*

$$H[X | Y] = \mathbb{E}_{p(x, y)}[-\log p(x | y)]. \quad (1.15)$$

Proposition 1.10 (Chain rule of entropy). *Let X and Y be two random variables with a joint probability distribution $p(x, y)$. The joint entropy equals:*

$$H[X, Y] = H[Y] + H[X | Y] = H[X] + H[Y | X]. \quad (1.16)$$

More generally, the joint entropy of $X_1, \dots, X_n$ distributed according to $p(x_1, \dots, x_n)$ is

$$H[X_1, \dots, X_n] = \sum_{i=1}^n H[X_i | X_{i-1}, \dots, X_1] \quad (1.17)$$

Proof. Note $\log p(x, y) = \log p(y) + \log p(x | y) = \log p(x) + \log p(y | x)$, taking expectation with respect to the joint $p(x, y)$ gives the result. Repeated application gives the more general result. $\square$

Definition 1.11 (Mutual information, MI). *Let X and Y be two random variables with joint probability distribution $p(x, y)$ and marginal distributions $p(x)$ and $p(y)$, respectively. The mutual information $I(X; Y)$ is the relative entropy between the joint and the product of marginals:*

$$I(X; Y) = KL(p(x, y) \| p(x)p(y)) = \mathbb{E}_{p(x, y)} \left[\log \frac{p(x, y)}{p(x)p(y)} \right]. \quad (1.18)$$

Proposition 1.12 (Some properties of MI).

- *Non-negativity:* $I(X; Y) \geq 0$ with equality if and only if X and Y are independent.
- *Symmetry:* $I(X; Y) = I(Y; X)$.
- *MI and entropy:* $I(X; Y) = H[X] + H[Y] - H[X, Y]$.
- *MI and conditional entropy:* $I(X; Y) = H[Y] - H[Y | X] = H[X] - H[X | Y]$.

Proof. Non-negativity follows from Gibbs' inequality 1.7: $I(X; Y) \geq 0$, with equality if and only if $p(x, y) = p(x)p(y)$, i.e. if X and Y are independent. For the remaining properties: note that we can rewrite the definition of $I(X; Y)$ as

$$I(X; Y) = \mathbb{E}_{p(x, y)} \left[\log \frac{p(x, y)}{p(x)p(y)} \right] = \mathbb{E}_{p(x, y)} \left[\log \frac{p(y)p(x | y)}{p(x)p(y)} \right] \quad (1.19)$$

$$= \mathbb{E}_{p(x, y)} [\log p(x | y)] - \mathbb{E}_{p(x, y)} [p(x)] \quad (1.20)$$

$$= H[X] - H[X | Y] \quad (\text{next two equalities follow from Proposition 1.10}) \quad (1.21)$$

$$= H[Y] - H[Y | X] \quad (1.22)$$

$$= H[X] + H[Y] - H[X, Y]. \quad (1.23)$$

$\square$

The mutual information is the reduction in the uncertainty of X due to knowledge of Y , as per Equation (1.21). By symmetry, it is also equal to the reduction in the uncertainty of Y due to knowledge of X , as per Equation (1.22). In other words, X gives us as much information about Y as Y gives about X .

Remark 1.13 (Conditioning reduces entropy). *From Equation (1.21) and by non-negativity of mutual information, we have $H[X] \geq H[X | Y]$.*

Remark 1.14 (Entropy $\iff$ Self-information). *The mutual information of a random variable X with itself is the entropy of X*

$$I(X; X) = H[X] - H[X | X] = H[X]. \quad (1.24)$$

Definition 1.15 (Conditional MI). *Let X, Y and Z be random variables with a joint probability distribution $p(x, y, z)$. The conditional mutual information of X and Y given Z is*

$$I(X; Y | Z) = KL(p(x, y, z) \| p(x | z)p(y | z)) = \mathbb{E}_{p(x, y, z)} \left[\log \frac{p(x, y | z)}{p(x | z)p(y | z)} \right] \quad (1.25)$$

Proposition 1.16 (Chain rule of MI). *Let $X_1, \dots, X_N, Y$ be random variables, then*

$$I(X_1, \dots, X_n; Y) = \sum_{i=1}^n I(X_i; Y | X_{i-1}, \dots, X_1) \quad (1.26)$$

Proof. Using the conditional entropy formulation of MI and applying the chain rule of entropy

$$I(X_1, \dots, X_n; Y) = H[X_1, \dots, X_n] - H[X_1, \dots, X_n | Y] \quad (\text{Equation (1.21)}) \quad (1.27)$$

$$= \sum_{i=1}^N H(X_i | x_{i-1}, \dots, X_1) - \sum_{i=1}^N H(X_i | x_{i-1}, \dots, X_1, Y) \quad (\text{Proposition 1.10}) \quad (1.28)$$

$$= \sum_{i=1}^n I(X_i; Y | X_{i-1}, \dots, X_1) \quad (1.29)$$

□

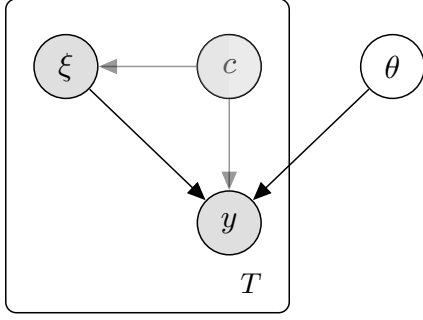


Figure 1.3: Graphical representation of the general class of Bayesian models we consider. The parameters θ are unobserved. Designs ξ , which are under our control, and outcomes y are observed. Contexts c (if present) can be deterministic or stochastic.

1.2.3 Bayesian experimental design (BED)

In the Bayesian approach to learning, as outlined in Section §1.2.1, the goal is to infer the parameters θ of a model $p(\theta)p(y|\theta, x)$ from data pairs of covariates and outcomes, $(x_i, y_i)_{i=1}^N$. In some cases, this data is passively received from external sources without any control over its generation. A typical example is financial data, where x could represent company financials (e.g., quarterly revenue, costs, profit), and y could be the stock return over the subsequent quarter. In such cases, the data is fixed, as variables like revenue, costs, and profit are inherently beyond our influence. This scenario is known as classical supervised learning.

However, in many scenarios, we have the ability to impact the data acquisition process, which can significantly enhance the utility of the data we collect for the purpose of learning θ . Examples include choosing questions to ask in a survey, selecting promising compounds for drug discovery trials, calibrating laboratory equipment for precise measurements, and choosing specific data points for annotation in machine learning tasks, to name a few.

We denote the covariates under our control as ξ , which we refer to as *experimental design variables* or simply *designs*. Depending on the field, these variables may also be called *actions* (in the bandits literature and reinforcement learning), *treatments* (in causal inference), or *controls* (in optimal control). If there are any elements that we cannot control, we denote them by c and refer to them as *context variables*, such that $\xi \cup c = x$. Contexts can be deterministic or stochastic and modelled separately. The general class of Bayesian models we consider, shown in Figure 1.3, factorises as $p(y, c, \xi, \theta) = p(\theta)p(c)p(y|\theta, \xi, c)$. For clarity of exposition we assume $\xi \equiv x$ throughout most of this thesis, so that the underlying model factorises as $p(y, \xi, \theta) = p(\theta)p(y|\theta, \xi)$. However, in general, the theory and methods we discuss are adaptable to scenarios that include contexts (e.g., Chapter §5).

Figure 1.4 illustrates how the choice of a design ξ can influence how much we learn about θ . Our initial uncertainty about θ is reflected in the prior $p(\theta)$, shown in Figure 1.4a. An experiment with design ξ_1 leads to an outcome y_1 , resulting in a posterior distribution $p(\theta|\xi_1, y_1)$ that

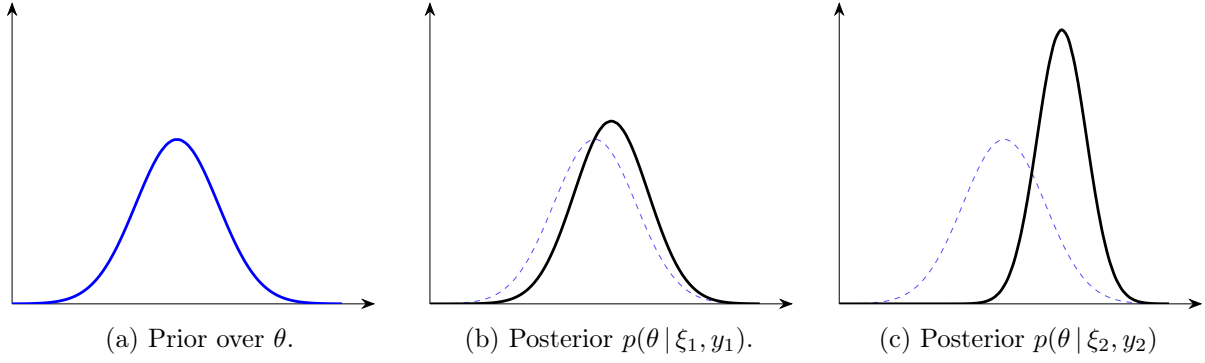


Figure 1.4: Hypothetical posteriors under two experiments with different designs and therefore outcomes. The posterior under the first design-outcome pair (ξ_1, y_1) in 1.4b is barely different from the prior and therefore not very informative. The posterior under the second design-outcome pair (ξ_2, y_2) in 1.4c is more peaked and therefore much more informative.

barely differs from the prior, indicating little new information was gained (1.4b). In contrast, an experiment using design ξ_2 with outcome y_2 , produce a considerably narrower, more peaked posterior distribution $p(\theta | \xi_2, y_2)$ (1.4c). We could therefore conclude that the data (ξ_2, y_2) is inherently *more informative* about θ than (ξ_1, y_1) , as it makes us more confident, or equivalently, less uncertain, in its estimate. This illustration intuitively guides us to what characterises an optimal experimental design—it is one that maximises information gain and, correspondingly, reduces our uncertainty regarding the unknown parameters of the model.

Information gain

Mathematically, we define the *information gain* as the difference between the Shannon entropy of the prior and that of the posterior.

Definition 1.17 (Information gain). *Given a model $p(\theta)p(y | \theta, \xi)$, the information gain about the model parameters θ of an experimental design ξ and outcome y is*

$$IG_\theta(\xi, y) = H[p(\theta)] - H[p(\theta | \xi, y)] = \mathbb{E}_{p(\theta | \xi, y)}[\log p(\theta | \xi, y)] - \mathbb{E}_{p(\theta)}[\log p(\theta)], \quad (1.30)$$

where $H[\cdot]$ denotes the Shannon entropy, defined in Equation (1.8).

Information gain vs relative entropy. Note that the entropy of the posterior distribution $p(\theta | \xi, y)$ can be less or greater than the entropy of the prior distribution $p(\theta)$. In other words, **the information gain can be positive or negative**, i.e. we have $IG_\theta(\xi, y) \in \mathbb{R}$. However, from the point of view of the posterior distribution, its **relative entropy** (or KL divergence)

(1.10) to the prior is always **non-negative**. Note that we can write

$$\text{IG}_\theta(\xi, y) = \text{KL}(p(\theta | \xi, y) \| p(\theta)) + (H[p(\theta)] - H[p(\theta | \xi, y) \| p(\theta)]), \quad (1.31)$$

and $H[p(\theta)] \neq H[p(\theta | \xi, y) \| p(\theta)]$, unless $p(\theta | \xi, y) = p(\theta)$. However, the two quantities are related; once averaged over all possible outcomes, y the two are in fact equivalent (1.37).

Expected Information Gain (EIG)

The information gain can be used to rank design-outcome *pairs* (ξ, y) , i.e. once the experiment ξ was been conducted and the outcome y was received. To select designs *before* observing y , we average over all possible outcomes we might observe if we ran that experiment. This gives us the Expected Information Gain (EIG, Lindley, 1956):

Definition 1.18 (Expected information gain (EIG)). *Given a model $p(\theta)p(y | \theta, \xi)$, the expected information gain about the model parameters θ of an experimental design ξ is*

$$\text{EIG}_\theta(\xi) = \mathbb{E}_{p(y|\xi)}[\text{IG}_\theta(\xi, y)] = H[p(\theta)] - \mathbb{E}_{p(y|\xi)}[H[p(\theta | \xi, y) \| p(\theta)]], \quad (1.32)$$

where $p(y | \xi) = \mathbb{E}_{p(\theta)}[p(y | \theta, \xi)]$ is the marginal distribution of the outcome under the model.

Expected information gain and expected relative entropy. As we saw, the difference between the prior and posterior entropies does not equal the relative entropy, $\text{IG}_\theta(\xi, y) \neq \text{KL}(p(\theta | \xi, y) \| p(\theta))$. If we take expectation with respect to $p(y | \xi)$ in both sides of Equation (1.31):

$$\mathbb{E}_{p(y|\xi)}[\text{IG}_\theta(\xi, y)] = \mathbb{E}_{p(y|\xi)}[\text{KL}(p(\theta | \xi, y) \| p(\theta))] + \mathbb{E}_{p(y|\xi)}[H[p(\theta)] - H[p(\theta | \xi, y) \| p(\theta)]] \quad (1.33)$$

Expanding the entropy and cross-entropy in the second expectation, and using the fact that under our model $p(y | \xi)p(\theta | \xi, y) = p(\theta)p(y | \theta, \xi)$

$$\mathbb{E}_{p(y|\xi)}[H[p(\theta)] - H[p(\theta | \xi, y) \| p(\theta)]] = \mathbb{E}_{p(y|\xi)p(\theta|\xi,y)}[\log p(\theta)] - \mathbb{E}_{p(y|\xi)p(\theta)}[\log p(\theta)] \quad (1.34)$$

$$= \mathbb{E}_{p(\theta)p(y|\theta,\xi)}[\log p(\theta)] - \mathbb{E}_{p(y|\xi)p(\theta)}[\log p(\theta)] \quad (1.35)$$

$$= \mathbb{E}_{p(\theta)}[\log p(\theta)] - \mathbb{E}_{p(\theta)}[\log p(\theta)] = 0. \quad (1.36)$$

In other words

$$\mathbb{E}_{p(y|\xi)}[\text{IG}_\theta(\xi, y)] = \mathbb{E}_{p(y|\xi)}[\text{KL}(p(\theta|\xi, y) \| p(\theta))]. \quad (1.37)$$

as required.

Expected information gain and mutual information. Again using the fact that under our model $p(y|\xi)p(\theta|\xi, y) = p(\theta)p(y|\theta, \xi)$, we can re-write the EIG as follows:

$$\text{EIG}_\theta(\xi) = \mathbb{E}_{p(y|\xi)}[\text{IG}_\theta(\xi, y)] = H[p(\theta)] - \mathbb{E}_{p(y|\xi)}[H[p(\theta|\xi, y)]] \quad (1.38)$$

$$= \mathbb{E}_{p(y|\xi)p(\theta|\xi, y)}[\log p(\theta|\xi, y)] - \mathbb{E}_{p(\theta)}[\log p(\theta)] \quad (1.39)$$

$$= \mathbb{E}_{p(\theta)p(y|\theta, \xi)}[\log p(\theta|\xi, y)] - \mathbb{E}_{p(\theta)}[\log p(\theta)] \quad (1.40)$$

$$= \mathbb{E}_{p(y, \theta|\xi)} \left[\log \frac{p(y, \theta|\xi)}{p(\theta)p(y|\xi)} \right] \quad (1.41)$$

$$= I(\Theta; Y | \xi). \quad (1.42)$$

In other words, the EIG of a design ξ equals the mutual information (MI) between the parameters θ and the outcome y under the design ξ . The next proposition formalises the various equivalent formulations of the expected information gain.

Proposition 1.19 (Equivalent EIG formulations). *The expected information gain can be expressed as*

$$\text{EIG}_\theta(\xi) = \mathbb{E}_{p(y|\xi)}[\text{IG}_\theta(\xi, y)] \quad \text{Definition (1.18)} \quad (1.43)$$

$$= \mathbb{E}_{p(y|\xi)}[\text{KL}(p(\theta|\xi, y) \| p(\theta))] \quad \text{expected KL formulation} \quad (1.44)$$

$$= I(\Theta; Y | \xi) = \mathbb{E}_{p(\theta, y|\xi)} \left[\log \frac{p(\theta, y|\xi)}{p(\theta)p(y|\xi)} \right] \quad \text{mutual information} \quad (1.45)$$

$$= H[\Theta] - H[\Theta | Y, \xi] = \mathbb{E}_{p(\theta, y|\xi)} \left[\log \frac{p(\theta|\xi, y)}{p(\theta)} \right] \quad \text{posterior formulation} \quad (1.46)$$

$$= H[Y | \xi] - H[Y | \Theta, \xi] = \mathbb{E}_{p(\theta, y|\xi)} \left[\log \frac{p(y|\theta, \xi)}{p(y|\xi)} \right] \quad \text{likelihood formulation} \quad (1.47)$$

$$= H[\Theta] + H[Y | \xi] - H[Y, \Theta | \xi] \quad \text{joint entropy formulation} \quad (1.48)$$

Proof. The expected KL formulation was derived in Equation (1.37). The conditional entropy and joint entropy formulations follow directly from Proposition (1.12) (to keep the notation consistent, we are denoting the random variables in capitals). $\square$

Remark 1.20 (Non-negativity of the EIG). *Since the EIG is equivalent to a mutual information, we have $\text{EIG}_\theta(\xi) \geq 0$ with equality if and only if θ and $y | \xi$ are independent.*

Remark 1.21 (EIG and conditional MI). *The EIG is **not** a conditional mutual information. Recall that conditional mutual information (Definition 1.15) quantifies the amount of information shared between two random variables conditioned on a third random variable. The conditional mutual information is an expectation over the joint distribution of the three random variables. In contrast, the EIG (Definition 1.18) whilst involving conditioning, is the mutual information between the parameters θ and the outcomes y given a specific, fixed experimental design ξ , not averaged over a distribution of ξ . It is an expectation over the joint distribution of the two random variables, conditional on the design $p(y, \theta | \xi)$.*

Remark 1.22 (EIG and BALD). *The likelihood formulation of Equation (1.47) is known as Bayesian Active Learning by Disagreement (BALD, Houlby et al., 2011) in the Bayesian active learning literature.*

Decision-theoretic design criteria

Whilst the EIG is the predominant criterion for design selection within the Bayesian experimental design framework (Foster, 2021; Rainforth et al., 2024), alternative choices are also possible. Lindley (1972) expanded upon the EIG criterion by developing a more general decision-theoretic approach that allows for the optimisation of an arbitrary *utility function*. The general recipe for optimising designs starts with a utility function that maps θ , designs ξ , and outcomes y to real numbers, $u : \theta, \xi, y \mapsto \mathbb{R}$. This function assigns a utility to the design-outcome pair (ξ, y) under the assumption that θ is the true parameter. Averaging over all possible parameters gives the expected utility for a given design-outcome pair (ξ, y) :

$$U(\xi, y) := \mathbb{E}_{p(\theta | \xi, y)}[u(\theta, \xi, y)]. \quad (1.49)$$

Averaging over all possible outcomes then gives a design criterion that can be optimised:

$$U(\xi) = \mathbb{E}_{p(y | \xi)}[U(\xi, y)]. \quad (1.50)$$

The EIG criterion is a specific instance within this more general framework, and can be derived by setting the utility function u to the log-ratio of the posterior and prior densities, which then gives an expected utility U equal to the relative entropy (KL divergence) between posterior and prior, as seen in Equation (1.37):

$$u(\theta, \xi, y) = \log \frac{p(\theta \mid \xi, y)}{p(\theta)} \quad U(\xi, y) = \text{KL}(p(\theta \mid \xi, y) \parallel p(\theta)) \quad U(\xi) = \text{EIG}_\theta(\xi) \quad (1.51)$$

This decision-theoretic framework enables the integration of highly specific objectives by selecting appropriate utility functions tailored to particular problems. However, developing suitable and robust utility functions presents considerable challenges. This difficulty contributes to the prevalent use of EIG as a design criterion. EIG is favoured not only for its simplicity and convenient mathematical properties but also because it consistently yields effective designs in practical applications.

Despite its appealing mathematical characteristics, the EIG has its drawbacks—it is notoriously difficult to estimate and optimise, especially in cases involving complex models with high-dimensional parameter and design spaces.

1.3 Literature review: Computational methods for BED

In this section, we review existing computational methods for estimating and optimising EIG, a central mathematical object in BED. The primary challenge in working with EIG is its *doubly intractable* nature (Rainforth et al., 2018; Zheng et al., 2018), a complexity that persists regardless of the formulation used (Proposition 1.19). The term “doubly intractable” refers to the presence of two levels of computational intractability:

1. **Outer level.** This involves computing an integral (expectation) over a function of some quantity. This expectation is difficult to compute due to the integration over complex distributions that are not straightforward to handle analytically. Standard Monte Carlo methods can be applied, but only when the inner quantity is tractable.
2. **Inner level.** The function within the integral depends on quantities that are themselves intractable and cannot be computed analytically—either the posterior distribution $p(\theta | \xi, y)$ (in 1.43, 1.44 and 1.46), or the Bayesian marginal distribution $p(y | \xi)$ (in 1.45, 1.47 and 1.48).

This means that estimating it for any given design ξ poses a significant challenge, which in turn makes the optimisation task of finding the optimal design $\xi^* = \arg \max_{\xi} \text{EIG}_{\theta}(\xi)$ even more computationally demanding. Initially, we focus on computational methods for single experiments before considering the additional complexities introduced by multiple experiments in a non-adaptive or adaptive manner.

1.3.1 EIG estimation

Although all EIG formulations from Proposition (1.19) are doubly intractable, some might be easier to work with, depending on the problem at hand, and assumptions we are willing to make.

Sample-based approaches

One possible approach is to use Monte Carlo to approximate the unknown distribution in the inner loop—the Bayesian marginal $p(y | \xi)$, or the posterior $p(\theta | \xi, y)$.

Nested (double-loop) Monte Carlo. Starting with the likelihood formulation from Equation (1.47), we can employ a nested Monte Carlo (NMC), also known as double-loop Monte Carlo (DLMC), estimator, using L samples to approximate the Bayesian marginal and N for

the outer expectation

$$\text{EIG}(\xi) \approx \mathcal{U}^{\text{NMC}}(\xi; L) := \mathbb{E}_{p(\theta)} \log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L} \sum_{\ell=1}^L p(y_n | \theta_{n,\ell}, \xi)}, \quad (1.52)$$

Its finite-sample Monte Carlo estimator then is:

$$\text{EIG}(\xi) \approx \hat{\mathcal{U}}^{\text{NMC}}(\xi; N, L) := \frac{1}{N} \sum_{n=1}^N \left[\log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L} \sum_{\ell=1}^L p(y_n | \theta_{n,\ell}, \xi)} \right], \quad (1.53)$$

where we sample $\theta_{n,0} \sim p(\theta)$, $y_n \sim p(y | \theta_{n,0}, \xi)$ and $\theta_{n,\ell} \sim p(\theta)$. This estimator has been widely studied and applied in the context of BED (Hamada et al., 2001; Cook et al., 2008; Ryan et al., 2014; Beck et al., 2018; Rainforth et al., 2018, among others).

However, this estimator has several limitations, including:

1. **Bias:** the estimator is biased (in fact it constitutes a lower bound). It is, however, consistent, i.e. $\hat{I}^{\text{NMC}}(\xi; N, L) \rightarrow \text{EIG}(\xi)$ as $N, L \rightarrow \infty$.
2. **Slow convergence:** its computational cost is $\mathcal{O}(NL)$, has an RMSE convergence rate $\mathcal{O}(N^{-1/2} + L^{-1})$, and overall rate $\mathcal{O}(T^{-1/3})$ when choosing $L \propto \sqrt{N}$, which is asymptotically optimal (Rainforth et al., 2018).
3. **Numerical instability:** The estimate of the Bayesian marginal distribution may suffer from arithmetic underflow if all likelihoods are exceedingly small, leading to a numerically zero denominator.
4. **Requires analytic likelihood:** to compute Equation (1.52) we need to be able to evaluate the likelihood, making the estimator unsuitable for implicit models.

Monte Carlo with Laplace approximation. Rather than approximating the Bayesian marginal, Monte Carlo with Laplace approximation (MCLA) approximates posterior distribution $p(\theta | \xi, y)$ *analytically* using a Gaussian centered at the maximum a posteriori (MAP) estimate, $\hat{\theta}$, and covariance, $\hat{\Sigma}(\hat{\theta})$, given by the inverse of the Hessian matrix at that MAP estimate (Papadimitriou, 2004; Long et al., 2013, 2015; Beck et al., 2018, among others). Beck et al. (2018) derive the following MCLA estimator:

$$\text{EIG}_{\theta}(\xi) \approx \hat{I}^{\text{MCLA}}(\xi; N) := \frac{1}{N} \sum_{n=1}^N \left(-\frac{1}{2} \log((2\pi)^d \det(\hat{\Sigma}(\theta_n))) - \frac{d}{2} - \log p(\theta_n) \right), \quad (1.54)$$

where $\theta_n \sim p(\theta)$. This method relies on an analytic likelihood and prior and further assumes that the posterior is Gaussian. Due to this strong assumption, MCLA tends to introduce significant bias and does not provide an asymptotically consistent estimator.

Nested Monte Carlo with Importance Sampling. To enhance the Nested Monte Carlo (NMC) approach and address some of its inherent limitations, particularly numerical stability and bias, incorporating importance sampling offers a viable solution. This method, referred to as Nested Monte Carlo with Importance Sampling (NMCIS), leverages an approximation $q(\theta)$ to the posterior $p(\theta | \xi, y)$, to guide the sampling process for a better estimate of the Bayesian marginal:

$$p(y | \xi) = \int p(\theta) p(y | \theta, \xi) d\theta = \int q(\theta) \frac{p(\theta) p(y | \theta, \xi)}{q(\theta)} d\theta \approx \frac{1}{L} \sum_{\ell=1}^L \frac{p(\theta_\ell) p(y | \theta_\ell, \xi)}{q(\theta_\ell)}, \quad (1.55)$$

which then gives the following EIG estimator:

$$\text{EIG}_\theta(\xi) \approx \hat{I}^{\text{NMCIS}}(\xi; N, L) := \frac{1}{N} \sum_{n=1}^N \left[\log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L} \sum_{\ell=1}^L w_{n,\ell} p(y_n | \theta_{n,\ell}, \xi)} \right], \quad (1.56)$$

where $w_{n,\ell} = \frac{p(\theta_{n,\ell})}{q(\theta_{n,\ell})}$ are the importance weights, $\theta_{n,0} \sim p(\theta)$, $y_n \sim p(y | \theta_{n,0}, \xi)$ and $\theta_{n,\ell} \sim q(\theta)$.

One practical choice for the proposal $q(\theta)$ is the Laplace approximation of the posterior discussed above, $q(\theta) = \mathcal{N}(\hat{\theta}, \hat{\Sigma}(\hat{\theta}))$. The statistical properties for this estimator, including its optimal settings, have been thoroughly studied, for example, in Beck et al. (2018).

Remark 1.23 (Analytic likelihood assumption). *Generally, all the sample-based approaches require the likelihood function to be analytic. When this is not the case, the typical workflow involves approximating the likelihood function as a separate step (e.g. Huan and Marzouk (2013) use a polynomial chaos surrogate) then applying the chosen sample-based estimator.*

Variational approaches

In recent years, variational methods utilising neural network-based estimators have gained increasing popularity in the field of Bayesian experimental design. This is largely due to their ability to effectively approximate complex distributions, a capability that has been greatly enhanced by advancements in machine learning and deep learning techniques (Foster et al., 2019; Kleinegesse and Gutmann, 2020; Foster et al., 2020; Kleinegesse and Gutmann, 2021).

Variational estimators produce functional approximations to one or more of the densities in-

volved in the computation of the EIG. By removing the inner level of intractability, variational approaches achieve a convergence rate of $\mathcal{O}(T^{-1/2})$, which is the same as conventional Monte Carlo estimators (Foster et al., 2019). However, this comes at the cost of introducing some bias. The good news is that these estimators often provide a lower or upper bound on the true EIG, allowing us to effectively sandwich it.

Although variational estimators introduce bias, there are ways to mitigate this issue. Sufficiently powerful approximators can minimise the magnitude of the bias. Moreover, hybrid approaches that integrate techniques like importance sampling can further reduce bias and, in some cases, even guarantee asymptotic consistency. Table 1.1 provides a summary of the variational estimators discussed below, along with their statistical properties.

Variational posterior. Suppose we approximate the mapping $y \rightarrow p(\theta | \xi, y)$ using a variational posterior $q_p(\theta | y, \xi)$ and use it in place of the true one posterior in Equation (1.46). This gives us a variational *lower bound* on the EIG (simple corollary of Gibbs inequality 1.7):

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{post}}(\xi, q_p) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{q_p(\theta | y, \xi)}{p(\theta)} \right] \quad (1.57)$$

which is tight if and only if $q_p(\theta | y, \xi) = p(\theta | y, \xi)$. The corresponding Monte Carlo estimator is

$$\mathcal{L}^{\text{post}}(\xi, q_p) \approx \hat{\mathcal{L}}^{\text{post}}(\xi, q_p; N) := \frac{1}{N} \sum_{n=1}^N \log \frac{q_p(\theta_n | y_n, \xi)}{p(\theta_n)} \quad \theta_n \sim p(\theta), y_n \sim p(y | \theta_n, \xi) \quad (1.58)$$

This bound was first introduced in the context of BED by Foster et al. (2019). It has been previously used for mutual information estimation (Barber and Agakov, 2003) and is also known as the Barber-Agakov (BA) bound. Note that this bound requires us to be able to evaluate the prior density only, i.e. it is applicable to models with *implicit likelihood* functions.

Remark 1.24 (Analytic prior assumption). *The variational posterior estimator equals the negative cross-entropy (1.9) between the joint and the approximate posterior, upto an additive constant, i.e. $\mathcal{L}^{\text{post}}(\xi, q_p) = -H(p(\theta, y | \xi) \| q_p(\theta_n | y_n, \xi)) + \text{const.}$ Therefore, if we are only interested in ranking designs (as opposed to their exact EIG), the analytic evaluation of the prior is actually not necessary.*

Variational marginal. Instead of approximating the posterior, we could learn $q_m(\cdot | \xi) : y \mapsto p(y | \xi)$, a variational approximation of the marginal likelihood. Using it in place of the true one

in Equation (1.47) yields a variational *upper bound* on the EIG

$$\text{EIG}_\theta(\xi) \leq \mathcal{U}^{\text{marg}}(\xi, q_m) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{p(y|\theta, \xi)}{q_m(y|\xi)} \right], \quad (1.59)$$

which is tight if and only if $q_m(y|\xi) = p(y|\xi)$. The corresponding Monte Carlo estimator is

$$\mathcal{U}^{\text{marg}}(\xi, q_m) \approx \hat{\mathcal{U}}^{\text{marg}}(\xi, q_m; N) := \frac{1}{N} \sum_{n=1}^N \log \frac{p(y_n|\theta_n, \xi)}{q_m(y_n|\xi)}, \quad \theta_n \sim p(\theta), y_n \sim p(y|\theta_n, \xi). \quad (1.60)$$

In the context of BED, the variational marginal estimator was proposed in Foster et al. (2019). Note that it is related to the NMC (DLMC) estimator (1.52), in that it provides a direct, closed-form approximation to the sampling-based sum in the denominator, i.e. $q_m(y|\xi) \approx \frac{1}{L} \sum_{\ell=1}^L p(y|\theta_\ell, \xi)$, $\theta \sim p(\theta_\ell)$. As with the NMC, this bound requires an *explicit likelihood* function.

Remark 1.25 (NMC gives an upper bound). *Although it does not give an analytic approximation, setting $q_m(y|\xi) = \frac{1}{L} \sum_{\ell=1}^L p(y|\theta_\ell, \xi)$ is a valid choice, meaning that the NMC estimator also constitutes an upper bound to the EIG.*

Variational marginal and likelihood. The variational marginal estimator can be easily modified to also accommodate implicit likelihood models. This involves introducing an additional density, $q_\ell(y|\theta, \xi)$, that serves as a surrogate for the unknown likelihood, leading to the following approximation

$$\text{EIG}_\theta(\xi) \approx \mathcal{I}^{\text{m}+\ell}(\xi, q_\ell, q_m) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{q_\ell(y|\theta, \xi)}{q_m(y|\xi)} \right], \quad (1.61)$$

and a corresponding finite-sample Monte Carlo estimator:

$$\mathcal{I}^{\text{m}+\ell}(\xi, q_\ell, q_m) \approx \hat{\mathcal{I}}^{\text{m}+\ell}(\xi, q_\ell, q_m; N) := \frac{1}{N} \sum_{n=1}^N \log \frac{q_\ell(y_n|\theta_n, \xi)}{q_m(y_n|\xi)}, \quad \theta_n \sim p(\theta), y_n \sim p(y|\theta_n, \xi). \quad (1.62)$$

The estimator was proposed in Foster et al. (2019) and *does not form an upper, nor a lower bound* on the EIG. The main advantage of this estimator is that it only requires samples from the prior distribution $p(\theta)$ and the observation model $p(y|\theta, \xi)$, therefore accommodating models where these distributions are only implicitly defined. This means that both of these densities can be implicitly defined, allowing for greater flexibility of the underlying models. However, since $\mathcal{I}^{\text{m}+\ell}$ is not an upper or a lower bound, quantifying its bias and evaluating the overall

reliability of its estimates can be particularly challenging. In particular, the estimator is not guaranteed to be consistent, and generally remains biased.

Variational nested Monte Carlo (VNMC). Recall that NMCIS (1.56) uses a proposal $q(\theta)$ to better approximate the Bayesian marginal distribution, with traditional approaches using a Laplace approximation (as in Beck et al., 2018, , for example). We could instead learn such a more flexible proposal distribution $q_v(\theta | y, \xi)$, and (hopefully better) approximate the marginal in a similar manner:

$$p(y | \xi) = \int p(\theta) p(y | \theta, \xi) d\theta = \int q_v(\theta | y, \xi) \frac{p(\theta) p(y | \theta, \xi)}{q_v(\theta | y, \xi)} d\theta \approx \frac{1}{L} \sum_{\ell=1}^L \frac{p(\theta_\ell) p(y | \theta_\ell, \xi)}{q_v(\theta_\ell | y, \xi)} \quad (1.63)$$

The variational NMC estimator (VNMC) estimator, proposed by Foster et al. (2019), forms a variational *upper bound* on the EIG and is given by:

$$\text{EIG}_\theta(\xi) \leq \mathcal{U}^{\text{VNMC}}(\xi, q_v; L) := \mathbb{E}_{p(\theta_0) p(y | \theta_0, \xi) q_v(\theta_{1:L} | y, \xi)} \left[\log \frac{p(y | \theta_0, \xi)}{\frac{1}{L} \sum_{\ell=1}^L \frac{p(\theta_\ell) p(y | \theta_\ell, \xi)}{q_v(\theta_\ell | \xi, y)}} \right]. \quad (1.64)$$

Its corresponding finite-sample MC estimator is:

$$\mathcal{U}^{\text{VNMC}}(\xi, q_v; L) \approx \hat{\mathcal{U}}^{\text{VNMC}}(\xi, q_v; N, L) := \frac{1}{N} \sum_{n=1}^N \left(\log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L} \sum_{\ell=1}^L w_{n,\ell} p(y_n | \theta_{n,\ell}, \xi)} \right), \quad (1.65)$$

where $w_{n,m} = \frac{p(\theta_{n,\ell})}{q(\theta_{n,\ell} | \xi, y_n)}$ are the importance weights, $\theta_{n,0} \sim p(\theta)$, $y_n \sim p(y | \theta_{n,0}, \xi)$, $\theta_{n,\ell} \sim q_v(\theta | \xi, y_n)$. The bound can generally be tightened in two ways: either by letting $L \rightarrow \infty$ or if $q_v(\theta | y_n, \xi) = p(\theta | y_n, \xi)$ for any $L \geq 1$. What this means is that in addition to asymptotic consistency, the finite sample estimator $\hat{I}^{\text{VNMC}}$ offers lower bias than the traditional NMC for the same number of inner samples L , as long as q_v approximates the posterior better than the prior. Note that VNMC requires an explicit likelihood function as well as an explicit prior.

Adaptive Contrastive Estimation (ACE). Foster et al. (2020) propose a simple modification of the VNMC estimator (1.64) that turns it into a variational *lower bound*. This is achieved by incorporating the initial sample used in the numerator $p(y | \theta_0, \xi)$ into the denominator, effectively *overestimating* the marginal $p(y | \xi)$ and thereby creating a lower bound:

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{ACE}}(\xi, q; L) := \mathbb{E}_{p(\theta_0) p(y | \theta_0, \xi) q(\theta_{1:L} | y, \xi)} \left[\log \frac{p(y | \theta_0, \xi)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(\theta_\ell) p(y | \theta_\ell, \xi)}{q(\theta_\ell | \xi, y)}} \right] \quad (1.66)$$

Its corresponding finite-sample MC estimator is given by

$$\mathcal{L}^{\text{ACE}}(\xi, q; L) \approx \hat{\mathcal{L}}^{\text{ACE}}(\xi, q; N, L) := \frac{1}{N} \sum_{n=1}^N \log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L+1} \sum_{\ell=0}^L w_{n,\ell} p(y_n | \theta_{n,\ell}, \xi)}, \quad (1.67)$$

where $w_{n,m} = \frac{p(\theta_{n,\ell})}{q(\theta_{n,\ell} | \xi, y_n)}$ are the importance weights, $\theta_{n,0} \sim p(\theta)$, $y_n \sim p(y | \theta_{n,0}, \xi)$, and $\theta_{n,\ell} \sim q_v(\theta | \xi, y_n)$. The term “contrastive” in ACE refers to the use of samples $\theta_{1:L}$, which serve as contrasts or negative samples to the “primary” or positive sample θ_0 —despite originating from the same distribution, these contrastive samples did not generate the observed $y \sim p(y | \theta_0, \xi)$. The “adaptive” part stems from the fact that the proposal distribution $q(\theta | y, \xi)$ can depend on (or be adapted to) the value of the design-outcome pair (ξ, y) .

Similar to VNMC, the ACE bound can be tightened in two ways: in the limit as $L \rightarrow \infty$ or for any $L \geq 1$ if $q(\theta | y, \xi) = p(\theta | y, \xi)$. Thus, $\mathcal{L}^{\text{ACE}}$ is asymptotically consistent and its finite MC estimator, $\hat{\mathcal{L}}^{\text{ACE}}$ tends to exhibit low bias as long as q reasonably approximates the true posterior. Note that the ACE bound requires an *explicit likelihood* function as well as an *explicit prior*.

Prior Contrastive Estimation (PCE). Choosing the prior as a proposal, i.e. setting $q(\theta | y, \xi) = p(\theta)$ gives a much simpler, albeit inherently less powerful, lower bound on the EIG:

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{PCE}}(\xi; L) := \mathbb{E}_{p(\theta_0)p(y|\theta_0,\xi)p(\theta_{1:L})} \left[\log \frac{p(y | \theta_0, \xi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(y | \theta_\ell, \xi)} \right]. \quad (1.68)$$

The bound is tight in the limit $L \rightarrow \infty$, meaning that the estimator is asymptotically consistent. Its corresponding finite-sample Monte Carlo estimator is given by

$$\mathcal{L}^{\text{PCE}}(\xi; L) \approx \hat{\mathcal{L}}^{\text{PCE}}(\xi; N, L) := \frac{1}{N} \sum_{n=1}^N \log \frac{p(y_n | \theta_{n,0}, \xi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(y_n | \theta_{n,\ell}, \xi)}, \quad (1.69)$$

where $\theta_{n,\ell} \sim p(\theta)$, $y_n \sim p(y | \theta_{n,0}, \xi)$. Foster et al. (2020) propose this as a simpler alternative to the ACE bound (1.66), referring to it as the “prior” contrastive estimator (PCE) due to the use of the prior as the proposal distribution. The PCE is the lower bound counterpart to the NMC estimator (1.52)—the two are identical, except for the inclusion of the “primary” sample in the estimation of the Bayesian marginal distribution in the denominator. Note that the PCE bound requires only an *explicit likelihood* function, i.e. the prior can be implicit.

Remark 1.26 (PCE is sample-based). *The PCE bound does not actually make any variational*

approximations. Since it was proposed as a special case of the ACE bound and has mainly been discussed in the context of variational BED, we include it here.

Likelihood-free ACE. For cases when the likelihood is not available in closed form, Foster et al. (2020) propose a likelihood-free version of the ACE bound. LF-ACE introduces a function, called a *critic*, that maps parameters and outcomes to real numbers $U : \theta, y \mapsto \mathbb{R}$. The bound holds for any such U and is given by:

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{LF-ACE}}(\xi, q, U; L) := \mathbb{E}_{p(\theta_0)p(y|\theta_0, \xi)q(\theta_{1:L}|y, \xi)} \left[\log \frac{\exp(U(\theta_0, y))}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(\theta_\ell) \exp(U(\theta_\ell, y))}{q(\theta_\ell|y, \xi)}} \right] \quad (1.70)$$

The finite-sample Monte Carlo estimator is given by:

$$\mathcal{L}^{\text{LF-ACE}} \approx \hat{\mathcal{L}}^{\text{LF-ACE}}(\xi, q, U; N, L) := \frac{1}{N} \sum_{n=1}^N \log \frac{U(\theta_{n,0}, y_n)}{\frac{1}{L+1} \sum_{\ell=0}^L w_{n,\ell} U(\theta_{n,\ell}, y_n)}, \quad (1.71)$$

where $w_{n,\ell} = \frac{p(\theta_{n,\ell})}{q(\theta_{n,\ell}|y_n, \xi)}$ are the importance weights, $\theta_{n,0} \sim p(\theta)$, $y_n \sim p(y|\theta_{n,0}, \xi)$, and $\theta_{n,\ell} \sim q_v(\theta|\xi, y_n)$. The bound can be tightened only for the *optimal* critic, which is equal to the likelihood upto an additive term, i.e. $U^*(\theta, y) = p(y|\theta, \xi) + c(y)$, where $c(y)$ is an arbitrary function of the outcome. With the optimal critic, the bound can be tightened in two ways: in the limit as $L \rightarrow \infty$ or for any $L \geq 1$ if $q(\theta|y, \xi) = p(\theta|y, \xi)$. Note that LF-ACE requires an *explicit prior*.

InfoNCE. We can simplify the LF-ACE bound by choosing $q(\theta|y, \xi) = p(\theta)$, which gives the InfoNCE bound popular in the representation learning literature (van den Oord et al., 2018)

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{InfoNCE}}(\xi, U; L) := \mathbb{E}_{p(\theta_0)p(y|\theta_0, \xi)p(\theta_{1:L})} \left[\log \frac{U(\theta_0, y)}{\frac{1}{L+1} \sum_{\ell=0}^L U(\theta_\ell, y)} \right]. \quad (1.72)$$

Its corresponding finite-sample Monte Carlo estimator is given by

$$\mathcal{L}^{\text{InfoNCE}}(\xi, U; L) \approx \hat{\mathcal{L}}^{\text{InfoNCE}}(\xi, U; N, L) := \frac{1}{N} \sum_{n=1}^N \log \frac{U(\theta_{n,0}, y_n)}{\frac{1}{L+1} \sum_{\ell=0}^L U(\theta_{n,\ell}, y_n)}, \quad (1.73)$$

where $\theta_{n,\ell} \stackrel{\text{iid}}{\sim} p(\theta)$, $y_n \sim p(y|\theta_{n,0}, \xi)$. As before, the optimal critic is $U^*(\theta, y) = p(y|\theta, \xi) + c(y)$, with c being an arbitrary function that doesn't depend on θ .

Compared to the LF-ACE, an advantage of the InfoNCE is that it does not require an analytic

prior, so it is *completely distribution-free*. A disadvantage is that the bound can be tightened only asymptotically as $L \rightarrow \infty$ and for the optimal critic U^* . Additionally, the bound cannot exceed $\log(L + 1)$ (Poole et al., 2019), meaning that $\hat{\mathcal{L}}^{\text{InfoNCE}}$ can exhibit significant bias when estimating large EIGs or when using small number of inner samples L .

NWJ. Kleingesse and Gutmann (2020) were the first to adapt the NWJ *lower bound* (Nguyen et al., 2010) for BED. Similar to the InfoNCE lower bound, NWJ is *completely distribution-free* and has been effectively used in representation learning contexts (Belghazi et al., 2018). For any critic function $U : \theta, y \mapsto \mathbb{R}$, the NWJ bound is given by

$$\text{EIG}_\theta(\xi) \geq \mathcal{L}^{\text{NWJ}}(\xi, U) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)}[U(\theta, y)] - \frac{1}{e} \mathbb{E}_{p(\theta)p(y|\xi)}[\exp(U(\theta, y))]. \quad (1.74)$$

The corresponding finite-sample Monte Carlo estimator for this bound is

$$\mathcal{L}^{\text{NWJ}}(\xi, U) \approx \hat{\mathcal{L}}^{\text{NWJ}}(\xi, U; N, M) := \frac{1}{N} \sum_{n=1}^N U(\theta_n, y_n) - \frac{1}{e} \frac{1}{M} \sum \exp(U(\theta_m, y_m)), \quad (1.75)$$

where $\theta_n \sim p(\theta)$, $y_n \sim p(y|\theta_n, \xi)$, $\theta_m \sim p(\theta)$, and $y_m \sim p(y|\theta, \xi)$. The bound is tight for the optimal critic $U^*(\theta, y) = \log \frac{p(y|\theta, \xi)}{p(y|\xi)} + 1$. This means that the finite sample estimator $\hat{\mathcal{L}}^{\text{NWJ}}$ is unbiased when using this optimal critic. Although this estimator is generally not guaranteed to be consistent—since there is no guarantee that we find the optimal critic U^* —the NWJ estimator should exhibit lower bias (particularly if we allow sufficiently flexible family of critics U).

Remark 1.27. *The NWJ bound (Nguyen et al., 2010) is a bound on a KL-divergence between two distributions and as such is more general-purpose than the InfoNCE (van den Oord et al., 2018), which is a bound on the mutual information.*

Choosing an estimator

Selecting an appropriate variational EIG estimator is largely driven by the specific application. This subsection outlines some of the key considerations.

1. **Model complexity and availability of analytic densities.** This is perhaps the most important factor. If our Bayesian model has an analytic prior and likelihood, estimators like the ACE or its upper bound counterpart VNMC, might be preferred as they provide consistent and low-bias estimates. If the prior is not available in closed form (this is usually

Estimator	Eq	Bound	Consistent	Bias	Implicit likelihood	Implicit prior
$\mathcal{L}^{\text{post}}(\xi, q)$	(1.57)	Lower	✗	High	✓	✗
$\mathcal{U}^{\text{marg}}(\xi, q)$	(1.59)	Upper	✗	High	✗	✓
$\mathcal{I}^{\text{m}+\ell}(\xi, q_\ell, q_m)$	(1.61)	NA	✗	High	✓	✓
$\mathcal{U}^{\text{VNMC}}(\xi, q)$	(1.64)	Upper	✓	Low	✗	✗
$\mathcal{U}^{\text{NMC}}(\xi)$	(1.52)	Upper	✓	Low	✗	✓
$\mathcal{L}^{\text{ACE}}(\xi, q; L)$	(1.66)	Lower	✓	Low	✗	✗
$\mathcal{L}^{\text{PCE}}(\xi; L)$	(1.68)	Lower	✓	High	✗	✓
$\mathcal{L}^{\text{LF-ACE}}(\xi, q, U; L)$	(1.70)	Lower	✓	Low	✓	✗
$\mathcal{L}^{\text{InfoNCE}}(\xi, U; L)$	(1.72)	Lower	✗	High	✓	✓
$\mathcal{L}^{\text{NWJ}}(\xi, U)$	(1.74)	Lower	✗	Low(er)	✓	✓

Table 1.1: **Overview of variational estimators of the Expected Information Gain (EIG)**, $\text{EIG}_\theta(\xi)$ (1.32). We assume generative model $p(\theta, y | \xi)$ that factorizes as a prior over parameters θ and an observation likelihood, i.e. $p(\theta)p(y | \theta, \xi)$ as given. Here, $\xi \in \mathbb{R}^d$ represents the design variables over which we wish to optimize. The various estimators may additionally optimize over parameterised densities, denoted by q , and/or a function U mapping parameters and outcomes to real numbers, $U : \theta \times y \mapsto \mathbb{R}$. $L \in \mathbb{N}_+$ is a fixed hyper-parameter that we generally don’t optimize over; unless the bound is already tight, increasing L leads to a tighter bound.

the case if we’ve already done one round of experiments and our posterior becomes the new prior, more details in §1.3.3). If a distribution-free approach is required, the options are generally not great, as all estimators require exponential sample complexity. The InfoNCE estimator is generally easier to work with due to its lower variance, but the NWJ estimator leads to somewhat lower bias.

- Bias and consistency.** As discussed, it is generally difficult to obtain unbiased EIG estimators. However, employing sandwich bounds where possible, such as combining ACE and VNMC or PCE and NMC, can effectively provide a comprehensive view of potential biases with no additional computational overhead.
- Computational resources.** Estimators that require a higher number of samples, such as NMC/PCE, may offer tighter, asymptotically consistent estimates, but as discussed, their convergence rate is slow. These estimators are suitable for situations where computational resources (during the experiments itself) are adequate and where the precision of the estimate is important. Variational estimators, such as variational marginal or variational posterior, converge at the regular MC rate. However, they usually introduce a bias and

may require substantial computational resources to accurately learn the density (which, however, can be done offline, before the experiment itself).

A summary of assumptions and properties of these estimators can be found in Table 1.1.

1.3.2 EIG optimisation

Having efficient estimators of the EIG is crucial but not sufficient to identify optimal designs; optimising EIG is equally essential. Traditionally, the estimation and optimisation of the EIG have been treated as two distinct, separate problems. This separation has often led to using gradient-free optimisation methods to tackle the inherent intractability of EIG. Examples include simple approaches like grid and enumeration (Han and Chaloner, 2004; van den Berg et al., 2003; Ryan, 2003), evolutionary algorithms (Hamada et al., 2001; Price et al., 2018), and Gaussian process surrogates (Overstall and McGree, 2020) and Bayesian optimisation (Kleinegesse and Gutmann, 2018; von Kügelgen et al., 2019; Foster et al., 2019).

Dividing the BED problem into two separate stages, estimation and optimisation, can sometimes be advantageous, as it allows for the independent modification and enhancement of each phase. However, this approach often leads to inefficiencies, as it can result in excessive computational effort being spent on precisely estimating the EIG for designs that are ultimately suboptimal. Moreover, gradient-free optimisation methods, which are often employed in the two-stage approach, do not scale well with increasing dimensionality, making them a poor choice for complex models with high-dimensional parameter and design spaces.

In response to these challenges, recent advancements have shifted towards more scalable gradient-based methods that *jointly estimate and optimise* variational lower bounds on the EIG (Foster et al., 2020; Kleinegesse and Gutmann, 2020; Foster et al., 2021; Ivanova et al., 2021; Blau et al., 2022). These methods simultaneously learn the variational densities q or functions U while optimising the design, thereby avoiding the inefficiencies associated with the two-stage approach. In this subsection, we review the key components required for the joint gradient-based estimation and optimisation of variational EIG bounds: the parameterisation of variational families and the reparameterisation trick.

Parameterising variational distributions

The variational EIG estimators, summarised in Table 1.1, require us to define a variational family $\mathcal{Q}$ and find the q that will optimise the bound. Finding the optimal q is crucial for obtaining a

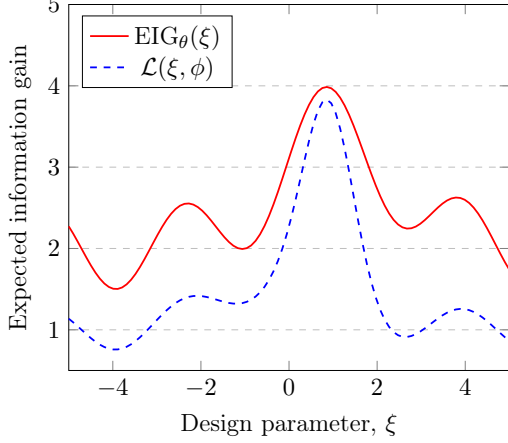


Figure 1.5: **Joint EIG estimation and optimisation.** The true EIG (solid red line) and its corresponding lower bound (dashed blue line) as functions of the design ξ . By maximising the variational lower bound $\mathcal{L}(\xi, \phi)$ simultaneously with respect to ξ and the variational parameters ϕ , we focus computational resources on the parts of the design space that are most promising. Specifically, the lower bound is relatively tight near the maximum of the true EIG, achieved at $x = 1$, ensuring that computational resources are not wasted on suboptimal designs.

tighter lower bound on the true EIG, which in turn leads to more effective optimisation of the design variables ξ . The variational family should be expressive enough to closely approximate the true distribution whilst remaining tractable for optimisation. Here, we briefly discuss possible parameterisations for $q \in \mathcal{Q}$, where q can represent a variational approximation to a variational approximation to a posterior $q(\theta \mid y, \xi)$, a marginal $q(y \mid \xi)$, or a likelihood $q(y \mid \theta, \xi)$. We generalise these cases as $q(\cdot \mid \mathcal{D})$, with $\mathcal{D}$ represents the respective combinations of θ , y , and ξ .

Assuming that $\theta \in \mathbb{R}^d$, a common choice for $\mathcal{Q}$ is a multivariate diagonal Gaussian distribution with parameters $\phi = (\mu, \text{diag}(\sigma^2))$. Here the parameters ϕ parameterise the variational family $\mathcal{Q}$. Often referred to as mean-field VI, this can be expressed as

$$q_\phi(\cdot \mid \mathcal{D}) = \mathcal{N}(\cdot; \mu, \text{diag}(\sigma^2)) \quad (1.76)$$

This choice of variational family is suitable if we expect the true distribution $p(\cdot \mid \mathcal{D})$ to have approximately normally distributed and uncorrelated components. However, mean-field VI may struggle to capture complex more complex distributions (e.g. multimodal) or correlations between the components, which can limit the quality of the EIG estimates and the effectiveness of the optimisation process.

It is important to note that different variational distributions q need to be learned for different $\mathcal{D}$. To improve efficiency and enable sharing of information across different $\mathcal{D}$, we can employ amortised variational inference (Kingma and Welling, 2014; Gershman and Goodman, 2014). In this approach, instead of using fixed constants for the parameters of the variational distribution, we learn functions $\mu(\mathcal{D})$ and $\sigma(\mathcal{D})$. We parameterise these by neural networks, so the variational distribution can be written as

$$q_\phi(\cdot \mid \mathcal{D}) = \mathcal{N}(\cdot; \text{NN}_{\phi_1}(\mathcal{D}), \text{diag}(\text{NN}_{\phi_2}(\mathcal{D}))) \quad \phi = (\phi_1, \phi_2). \quad (1.77)$$

As before, the variational family $\mathcal{Q}$ is parameterised by the neural network parameters ϕ . By leveraging neural networks to learn a mapping from data to the parameters of the variational distribution, amortised variational inference allows us to learn a function that maps data to distributions, rather than fitting a separate posterior for each individual dataset. This weight-sharing approach can significantly improve the efficiency of the optimisation process, as the learned mapping can be reused for different datasets without the need to learn a new variational distribution from scratch.

For more flexible ways to parametrise $\mathcal{Q}$, one might consider employing more expressive families, for example, full covariance Gaussian distributions. Another popular approach is Normalizing flows (Rezende and Mohamed, 2015), which are a class of invertible transformations, parameterised by (invertible) neural networks, that are used to transform a simple base distribution, such as a standard normal, into a more complex target distribution. By conditioning the parameters of the normalizing flow on the input data $\mathcal{D}$, we can learn a highly flexible and expressive variational families. This class of models, known as conditional normalizing flows (Winkler et al., 2019; Ardizzone et al., 2019), have been successfully employed in the context of amortised neural likelihood and/or posterior estimation (Radev et al., 2020; Papamakarios et al., 2021; Radev et al., 2023).

Once we have selected a variational family we can use stochastic gradient ascent to learn (near) optimal parameter values ϕ^* . The details of this procedure are discussed in the next subsection.

Remark 1.28 (Optimising critics). *Some of the variational bounds on EIG, such as the InfoNCE (1.72) and NWJ (1.74), involve a critic function $U : \theta, y \mapsto \mathbb{R}$ instead of variational distributions. This critic is typically unnormalised and can be viewed as a learnable discriminator that distinguish between samples from the joint distribution $p(\theta, y | \xi)$ and the product of marginals $p(\theta)p(y | \xi)$. Similar to the variational parameters, we parameterise the critic with a neural network with parameters ϕ and optimise U_ϕ jointly along with the designs and potentially other variational distributions using gradient-based methods.*

Stochastic optimisation and the reparameterisation trick

We provide the generic recipe for optimising the variational EIG estimators (Table 1.1) using stochastic gradient ascent (SGA) schemes (Robbins and Monro, 1951).

The bounds in Table 1.1 are generally defined as $\mathcal{L}(\xi, \phi) = \mathbb{E}_{p(\theta)f(y, \xi, \phi | \theta)}[g(\theta, y, \xi, \phi)]$, where ξ are the designs and ϕ are neural network parameters of variational families or critics; for NMC of Equation (1.52) and PCE of Equation (1.68) we have $\phi = \emptyset$. The function f is stochastic

and simulates the outcome y ; the function g varies across different bounds and may involve known densities, variational densities and/or critics. For example, for the InfoNCE bound in Equation (1.72), we have:

$$f(y, \xi, \phi | \theta) = p(y | \theta, \xi) p(\theta_{1:L} | y) \quad (1.78)$$

$$g(\theta, y, \xi, \phi) = \log \frac{U_\phi(\theta, y)}{\frac{1}{L+1} \sum_{\ell=1}^L (U_\phi(\theta_\ell, y) + U_\phi(\theta, y))} \quad (1.79)$$

As usual, we estimate $\mathcal{L}(\xi, \phi)$ using a finite N -sample Monte Carlo estimator, denoted as $\hat{\mathcal{L}}(\xi, \phi; N)$:

$$\mathcal{L}(\xi, \phi) \approx \hat{\mathcal{L}}(\xi, \phi; N) := \frac{1}{N} \sum_{n=1}^N g(\theta_i, y_i, \xi, \phi), \quad \text{where } \theta_i \sim p(\theta), y_i \sim f(y, \xi, \phi | \theta). \quad (1.80)$$

To optimise $\mathcal{L}(\xi, \phi)$, we need to compute its gradient $\nabla_{\xi, \phi} \mathcal{L}(\xi, \phi) = \nabla_{\xi, \phi} \mathbb{E}_{p(\theta) f(y, \xi, \phi | \theta)} [g(\theta, y, \xi, \phi)]$, accounting for the fact that the parameters over which we optimise appear in the distributions with respect to which expectations are taken. To deal with this issue we apply the *reparameterisation trick* (Mohamed and Rezende, 2015; Rezende et al., 2014; Kingma and Welling, 2014), also known as the *law of the unconscious statistician* among (conscious) statisticians.

The reparameterisation trick involves introducing a new random variable $\epsilon \sim p(\epsilon)$ with a fixed distribution, e.g. a standard normal, and defining a deterministic transformation $T(\epsilon, \xi, \phi)$ that maps ϵ to the desired distribution of y . When this is possible the variable y is called *reparametrisable* (most continuous distributions are reparametrisable). This allows us to rewrite the loss function in terms of an expectation that does not depend on the optimisable parameters

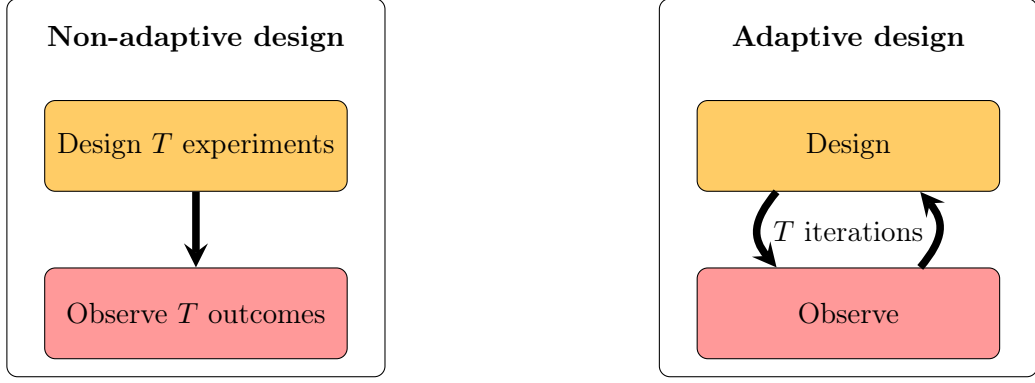
$$\mathcal{L}(\xi, \phi) = \mathbb{E}_{p(\theta) p(\epsilon)} [g(\theta, T(\epsilon, \xi, \phi), \xi, \phi)]. \quad (1.81)$$

We can then freely swap the order of the expectation and the gradient operators and obtain an *unbiased* estimate of the expectation using a simple Monte Carlo:

$$\nabla_{\xi, \phi} \mathcal{L}(\xi, \phi) = \mathbb{E}_{p(\theta) p(\epsilon)} [\nabla_{\xi, \phi} g(\theta, T(\epsilon, \xi, \phi), \xi, \phi)] \approx \nabla_{\xi, \phi} \hat{\mathcal{L}}(\xi, \phi; N) \quad (1.82)$$

Standard stochastic optimisation schemes, such as Adam (Kingma and Ba, 2014) or AdamW (Loshchilov and Hutter, 2017), can be employed to find $\xi^*, \phi^* \approx \arg \max_{\xi, \phi} \mathcal{L}(\xi, \phi)$. Our optimal design is ξ^* and an N -sample estimate of the EIG it achieves is given by $\hat{\mathcal{L}}(\xi^*, \phi^*; N) \approx \text{EIG}_\theta(\xi^*)$.

Remark 1.29 (Reparameterisation in practice). *Reparameterisation is handled automatically*



(a) Non-adaptive, static, batch or open-loop

(b) Adaptive, sequential or closed loop

Figure 1.6: The two classes of design strategies for multiple experiments.

in modern auto-differentiation software and deep learning frameworks such as PyTorch (Paszke et al., 2019) or TensorFlow (Abadi et al., 2016). In practice this means that as long as a distribution is reparametrisable (e.g. in PyTorch `torch.distributions.<D>.has_rsample` is `True`), there is no need to manually find and implement the transformation T , as all necessary computations are handled internally.

1.3.3 Multiple experiments

Throughout our discussion so far, we focused on designing a *single* experiment, denoted by ξ to learn about the parameters θ in a Bayesian model $p(\theta)p(y|\theta, \xi)$. However, in many cases, we need to conduct multiple experiments in order to gather sufficient information or data. There are two main approaches to designing these multiple experiments: static and sequential (Figure 1.6).

In the static approach, also known as *batch* or *open-loop* approach, all of the experiments are designed and conducted in parallel at once. This means that each experiment is planned independently of the others. In contrast, the sequential approach, also known as *adaptive* or *closed-loop* approach, involves designing and conducting experiments in a dynamic, iterative manner. In this approach, the results of each experiment are used to inform the design of the next experiment, allowing the experimenter to adapt to new information and optimise the experiments over time. This approach allows for more efficient use of resources and can potentially yield better results than the static approach.

In this subsection, we consider the scenario where we want to perform T experiments with designs $\xi_1, \dots, \xi_T$. We will refer to the sequence of design-outcome pairs from these T experiments as the *history*, and we will denote it by $h_T := \{(\xi_i, y_i)\}_{i=1}^T$.

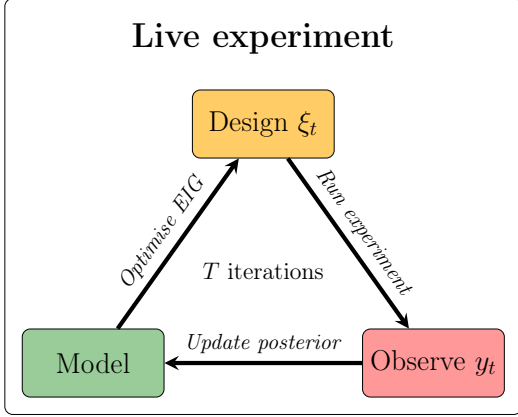


Figure 1.7: **Traditional adaptive BED lifecycle.** We obtain the design ξ_t by optimising the expected information gain about the underlying model parameters θ , EIG_θ . An experiment with design ξ_t is then conducted to obtain an outcome y_t . The pair (ξ_t, y_t) is added to the existing data of experimental histories $h_t = (\xi_t, y_t) \cup h_{t-1}$, with $h_0 = \emptyset$, and the underlying model is updated by fitting the posterior $p(\theta | h_t)$. This cycle is repeated for T iterations.

Non-adaptive experimentation

Instead optimising the expected information gain with respect to a single design, we can define a vector of designs $\boldsymbol{\xi} = (\xi_1, \dots, \xi_T)$, where each element ξ_i represents the design of the i -th experiment in the batch. We can then estimate and optimise $\text{EIG}_\theta(\boldsymbol{\xi})$ using any of the methods described in the previous subsection to obtain an optimal set of designs for the experiments in the batch.

Once the experiment is completed and we have obtained the experimental history h_T , we can use our favourite Bayesian inference method to obtain the posterior distribution

$$p(\theta | h_T) = \frac{p(\theta)p(y_{1:T} | \theta, \xi_{1:T})}{p(y_{1:T} | \xi_{1:T})} \quad (1.83)$$

This posterior distribution represents our updated beliefs about the values of our unknown parameters of interest, given the information we have obtained from the experiment.

There are scenarios in which static design may be the most appropriate choice for conducting experiments (Zanette et al., 2021; Matsushima et al., 2020). For instance, static design may be useful when experiments need to be performed in parallel, such as when multiple agents are collecting data asynchronously. In addition, static design may be useful when it is impractical to run adaptive experiments, due to engineering challenges or other limitations. For example, if an experiment involves a complex physical system that is difficult to control or manipulate, a static design may be more feasible than an adaptive one.

Often however, static design strategies are selected over adaptive ones due to the computational cost of traditional sequential BED methods, which we introduce next.

Traditional adaptive BED

In practice, it is often possible to conduct multiple experiments in sequence rather than in parallel. In these situations, allowing the design of each experiment ξ_t to be influenced by the data collected in previous experiment iterations $h_{t-1} := \{(\xi_i, y_i)\}_{i=1}^{t-1}$ can make the experiment less wasteful and more efficient at gathering useful information.

The traditional approach to designing adaptive experiments is a three-stage procedure illustrated in Figure 1.7. To find the current optimal design ξ_t we first fit the posterior distribution of the unknown parameters based on h_{t-1} —the experimental data collected so far

$$p(\theta | h_{t-1}) \propto p(\theta)p(y_{1:t-1} | \theta, \xi_{1:t-1}). \quad (1.84)$$

In the second stage, we select the optimal experimental design by maximizing the EIG using the updated posterior distribution as our new prior

$$\xi_t = \arg \max_{\xi} \text{EIG}_{\theta}^{h_{t-1}}(\xi), \quad (1.85)$$

$$\text{EIG}_{\theta}^{h_{t-1}}(\xi) := \mathbb{E}_{p(\theta | h_{t-1})p(y_t | \theta, \xi, h_{t-1})} \left[\log \frac{p(y_t | \theta, \xi, h_{t-1})}{p(y_t | \xi, h_{t-1})} \right]. \quad (1.86)$$

We refer to $\text{EIG}_{\theta}^{h_{t-1}}$ as the *one-step* or *incremental* EIG. In the third and final step we conduct an experiment with design ξ_t and obtain an outcome y_t . The design-outcome is added to the history $h_t = h_{t-1} \cup (\xi_t, y_t)$ and the process repeats.

There are two major limitations with this approach: its myopic nature and significant computational cost involved.

Computational cost. Estimating the posterior $p(\theta | h_t)$ in Equation (1.84), and the subsequent EIG optimisation in Equation (1.86) makes it impractical for long sequences of experiments or scenarios requiring rapid decision-making, such as those involving human participants. Additionally, even when we employ asymptotically exact inference methods such as HMC, in practice we only obtain approximate posteriors due to finite computational budget. Even small errors in these approximations can propagate through the EIG optimisation and ultimately lead to suboptimal designs.

Myopic nature. This one step ahead *greedy* or *myopic* design strategy focuses on optimising the next best choice of ξ without considering the potential impact of future experiments. Efforts to mitigate this limitation include non-myopic strategies such as those proposed in Huan and

Marzouk (2016), which use approximate dynamic programming, or techniques from the field of Bayesian optimisation (González et al., 2016; Jiang et al., 2020). However, these approaches still rely on posterior approximations and involve significant additional computational costs.

In order to run experiments in (near) real-time, one typically had to either abandon adaptivity or Bayesian optimality in their experimental design. In Chapters §2 and §3, we will see that it is actually possible to overcome these issues by learning a design policy upfront, before experimentation begins. This policy can then be deployed to make design decisions based on the previously gathered data, avoiding any posterior computations. This policy-based approach will enable us to make non-myopic design decisions in real time.

Illustrative examples revisited

We revisit the examples from the previous section to demonstrate some applications of actively acquiring data. We also highlight some of the key challenges that arise when designing adaptive experiments, and that will be addressed throughout this thesis.

Example 1.30 (BED for a Bayesian linear regression model). *Suppose we can choose the values of all the covariates $\mathbf{x}$, so that our designs $\xi \equiv x$. Let $\Xi = (\xi_1, \dots, \xi_N)$ denote our design matrix and $\mathbf{y} = (y_1, \dots, y_N)$. As already established, the posterior is also Gaussian*

$$p(\theta | \Xi, \mathbf{y}) = \mathcal{N}(\mu_\theta^{post}, \Sigma_\theta^{post}), \quad \mu_\theta^{post} = \sigma_\epsilon^{-2} \Sigma_\theta^{post} \Xi^\top \mathbf{y} \text{ and } \Sigma_\theta^{post} = (\sigma_\epsilon^{-2} \Xi^\top \Xi + (\Sigma_\theta^{prior})^{-1})^{-1}. \quad (1.87)$$

Note that the posterior variance matrix does not depend on the outcome vector $\mathbf{y}$. The entropy of a d -dimensional Gaussian $\mathcal{N}(z; \mu, \Sigma)$ is

$$H[Z] = \frac{1}{2} \log \det(\Sigma) + \frac{d}{2} (1 + \log(2\pi)). \quad (1.88)$$

The EIG of the design matrix Ξ therefore is:

$$EIG_\theta(\Xi) = H[p(\theta)] - \mathbb{E}_{p(\mathbf{y} | \Xi)} [H[p(\theta | \Xi, \mathbf{y})]] \quad (1.89)$$

$$= \frac{1}{2} \log \det(\Sigma_\theta^{prior}) + \frac{d}{2} (1 + \log(2\pi)) - \mathbb{E}_{p(\mathbf{y} | \Xi)} \left[\frac{1}{2} \log \det(\Sigma_\theta^{post}) + \frac{d}{2} (1 + \log(2\pi)) \right] \quad (1.90)$$

$$= \frac{1}{2} \log \det(\Sigma_\theta^{prior}) - \frac{1}{2} \log \det(\Sigma_\theta^{post}). \quad (1.91)$$

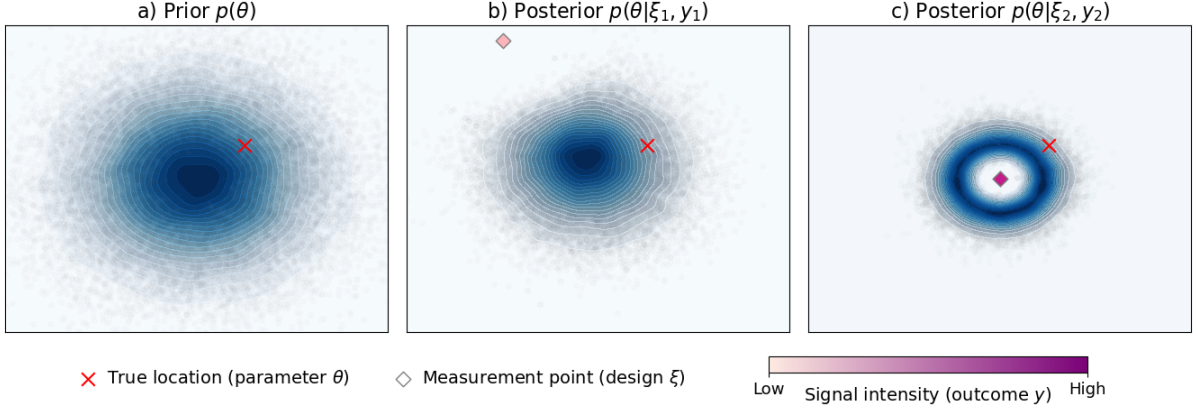


Figure 1.8: **Hidden object location finding.** The red cross marks the true but unknown location of the object, θ , representing our parameter of interest. Measurement point ξ is marked by a diamond, with its fill colour corresponding to the observed signal intensity. **Panel a):** The prior distribution $p(\theta)$ reflects our initial uncertainty about the location of the object, before any data is acquired. **Panel b):** The posterior $p(\theta | \xi_1, y_1)$ after observing y_1 at measurement location ξ_1 remains similar to the prior, suggesting that the design-outcome pair (ξ_1, y_1) provided limited new information. **Panel c):** The posterior $p(\theta | \xi_2, y_2)$ after observing y_2 at measurement location ξ_2 is significantly narrower, indicating that the acquired design-outcome pair (ξ_2, y_2) is notably more informative. Posteriors estimated with HMC.

Maximizing the EIG is therefore equivalent to minimizing $\det(\Sigma_\theta^{post})$, so the optimal design is

$$\Xi^* = \arg \min_{\Xi} \det(\sigma_\epsilon^{-2} \Xi^\top \Xi + (\Sigma_\theta^{prior})^{-1})^{-1}, \quad (1.92)$$

and can be found using standard optimisation techniques.

It is important to note that the model does not benefit from adaptivity, as the optimal design Ξ^* does not depend on the observed outcomes $\mathbf{y}$. Put differently, even if we were to design experiments sequentially, the design at each step would not be influenced by the previous outcomes.

Example 1.31 (BED for hidden object location finding). As previously discussed, the location finding model has a simple Gaussian likelihood, but a complicated intractable posterior. We use this model as a more realistic example, compared to the illustrative one in Figure 1.4, to demonstrate how different design choices (in this case measurement location) impacts the informativeness of the data we end up collecting. Figure 1.4a) shows the prior distribution $p(\theta)$, which reflects our initial uncertainty about the object's location. The true location of the object, marked by a red cross and sampled from that prior $\theta \sim p(\theta)$, is our target for discovery. Observing a signal intensity $y = y_1$ at this location leads to the posterior shown in Figure 1.4b). This posterior remains similar to the prior, indicating that the data (ξ_1, y_1) provides limited new information. Figure 1.4c) shows the posterior distribution after a measurement is taken at location $x = \xi_2$, which corresponds to the most likely location, according to our prior (and the

location that maximises EIG). This posterior is notably narrower and non-Gaussian. The data (ξ_2, y_2) is more informative than (ξ_1, y_1) , as it reduces uncertainty about θ more effectively.

Unlike the simple linear regression of the last example (where the optimal designs did not depend on the observed outcome), we can intuitively see that employing an adaptive strategy in choosing measurement location can be very beneficial. However, performing the posterior updates after each new measurement is taken will be very computationally costly and, for example, impractical to do in real time. These challenges are addressed in Chapter §2.

Example 1.32 (BED for stochastic SIR model). Recall that the Susceptible-Infected-Recovered (SIR) model is a widely used epidemiological model that describes the spread of an infectious disease through a population. In the context of BED, the aim is to estimate these two model parameters, infection rate β and recovery rate γ , by observing state populations $\mathbf{Y}(\tau)$ at particular measurement times τ , which serve as the experimental design variables. The SIR model has been studied extensively in the context of BED, for example, in Kleinegesse and Gutmann (2018); Dehideniya et al. (2018); Kleinegesse et al. (2020). These works highlight the challenges and complexities involved in working with implicit models. The lack of an analytic likelihood function makes the problem quite hard as conventional Monte Carlo methods (such as HMC used in the previous example) are not readily applicable. The core problem of performing quick and adaptive design decisions with implicit models is addressed in depth in Chapter §3.

1.4 Beyond parameter learning: Goal-oriented Bayesian experimental design

Conventional Bayesian experimental design chooses designs ξ that maximise the expected information gain $\text{EIG}_\theta(\xi)$ about the parameters θ of a generative model $p(\theta, y \mid \xi)$. This approach is equivalent to uniformly minimising the uncertainty of all the components of the parameters θ , as measured by Shannon entropy. However, in many practical applications, the primary objective may not be to necessarily reduce uncertainty about all parameters. Instead, the focus might be on a specific subset of parameters or on a distinct quantity of interest z which is derived from the parameters, such as $z = f_z(\theta)$. These quantities are often more directly relevant to the downstream objectives of the experiment, such as decision-making or predictive performance.

To address these needs, goal-oriented BED (Attia et al., 2018; Wu et al., 2021; Zhong et al., 2024) shifts the focus towards learning about such quantities of interest (QoI). These QoIs may be deterministic or stochastic functions of the model parameters θ . To encompass both deterministic and stochastic QoIs, we introduce the notation $p(z \mid \theta)$, which represents a proper probability distribution if f_z is stochastic, and a Dirac delta function $\delta(z - f_z(\theta))$ if f_z is deterministic. We assume that the QoI are conditionally independent of the outcomes y given the full set of model parameters, i.e. $p(z \mid y, \theta) = p(z \mid \theta)$.

Illustrative examples

Example 1.33 (Nuisance parameters). *As another example, suppose that $\theta = (\theta_{\text{target}}, \eta)$, where η is a set of nuisance parameters, that are of no interest to our learning objectives. In such cases, the QoI simplifies to $z = \theta_{\text{target}}$. This approach is particularly relevant in causal inference settings, where we may wish to learn conditional average treatment effects and treat the rest of the model parameters as nuisance variables.*

Example 1.34 ((Contextual) Bayesian optimisation). *Bayesian optimisation (BO) (see Snoek et al., 2012, for a review) considers the problem of finding the global maximiser of an unknown, expensive to evaluate objective function*

$$\xi^* = \arg \max_{\xi \in \Xi} f(\xi), \quad (1.93)$$

which is modelled probabilistically and assigned a prior $p(f)$. We can obtain a (noisy) measurement of f at a selected point ξ , i.e. $y \mid f, \xi \sim p(y \mid f(\xi))$.

We can introduce variables that we cannot control, but affect the outcomes y . We refer to these as contexts c , which can be known in advance or treated as stochastic, coming from some underlying distribution $p(c)$. The overall data generating process then becomes:

$$f \sim p(f) \quad c \sim p(c) \quad y | f, c, \xi \sim p(y | f(\xi), c) \quad (1.94)$$

(Contextual) BO can naturally be formulated within the BED framework. The only missing piece is identifying the quantity of interest. One possibility is to try to learn f uniformly. However, recall that our explicit goal is to find the $\arg \max$ of f and not the entire function! Specifically, any information about f in regions where it is way below its maximum is not going to be useful. Possible goal-oriented QoI include $z = \arg \max_{\xi \in \Xi} f(\xi)$ (Hernández-Lobato et al., 2014) and $z = \max_{\xi \in \Xi} f(\xi)$ (Wang and Jegelka, 2017). These are explored in detail in Ivanova et al. (2023), presented in §5 of this thesis.

Example 1.35 (BED for active causal discovery). Active causal discovery aims to uncover the causal structure of a system by performing interventions and observing their effects (Toth et al., 2022; Tigas et al., 2022; Annadani et al., 2023). The goal is to efficiently learn the underlying causal Directed Acyclic Graph (DAG), $g = (\mathbf{V}, E)$, where $\mathbf{V}$ represents the set of vertices (variables), E represents the set of directed edges indicating causal relationships, and $\mathbf{X}_v$ are the random variables indexed by $\mathbf{V}$. Additionally, it seeks to learn the parameters of mechanisms that describe how the direct causes affect each variable $\mathbf{X}_v$.

In a Structural Causal Model (SCM Peters et al., 2017), each variable X_i has an associated structural equation, which is a deterministic function of its direct causes $X_{pa(i)}$:

$$X_i = f_i(X_{pa(i)}, \epsilon_i; \theta), \quad i \in \mathbf{V}, \quad (1.95)$$

where $f_i(\cdot; \theta)$ are causal mechanisms, parameterised by θ , and ϵ_i are exogenous noise variables. An experiment in causal discovery corresponds to selecting the intervention target vertices and their corresponding states to set those targets. In other words, the experimental design ξ represents the choice of interventions to perform, specifying which variables to intervene on and their respective intervention values, and the goal is to gain information about the graph g , as well as the parameters θ of the causal mechanisms $f_i(\cdot; \theta)$.

The goal is to gain information about the graph and the mechanism parameters $z = (g, \theta)$. Recent work by Annadani et al. (2023)¹ explores differentiable multi-target causal BED, enabling the

¹Full citation: Annadani, Y., Tigas, P., **Ivanova, D. R.**, Jesson, A., Gal, Y., Foster, A., & Bauer, S.

optimisation of interventions for efficient causal discovery. The approach leverages gradient-based optimisation to find optimal interventions, taking into account the specific goals of learning the causal DAG and mechanism parameters.

1.4.1 EIG formulation and estimation for goal-oriented BED

Our generative model is still given by $p(\theta)p(y \mid \theta, \xi)$, but the objective is to gain information about the QoI z . To formulate the appropriate EIGs, we first write down the relevant distributions on z , expressing them as expectations to facilitate sample-based approximations.

Prior. Using the law of total probability, we marginalise the joint (θ, z) over θ and get:

$$p(z) = \int p(\theta, z) d\theta = \int p(\theta) p(z \mid \theta) d\theta = \mathbb{E}_{p(\theta)}[p(z \mid \theta)] \quad (1.96)$$

Likelihood. We need to integrate over all possible θ that are compatible with z . Again, using the law of total probability, we marginalise over θ to get

$$p(y \mid z, \xi) = \int p(\theta \mid z) p(y \mid \theta, z, \xi) d\theta = \int p(\theta \mid z) p(y \mid \theta, \xi) d\theta = \mathbb{E}_{p(\theta \mid z)}[p(y \mid \theta, \xi)], \quad (1.97)$$

where $p(\theta \mid z)$ is the posterior of θ given z . Note that this posterior can be complicated to obtain if f_z is stochastic or complicated. If f_z is *simple*, for example, deterministic and invertible, then $p(\theta \mid z) = \delta(\theta - f_z^{-1}(z))$ and we have $p(y \mid z, \xi) = p(y \mid \theta, \xi)$. Other situations, where $p(\theta \mid z)$ might be easy is in the case of nuisance parameters (Example 2), where $p(\theta \mid z) = p(\eta)$.

Joint. To get the joint on QoIs and observations, we need to marginalise over θ . Using the law of total probability and the fact that $y \perp\!\!\!\perp z \mid \theta$, we have:

$$p(z, y \mid \xi) = \int p(\theta, z, y \mid \xi) d\theta = \int p(\theta) p(z \mid \theta) p(y \mid \theta, \xi) d\theta = \mathbb{E}_{p(\theta)}[p(z \mid \theta) p(y \mid \theta, \xi)] \quad (1.98)$$

Information gain. As with conventional BED, the information gain about z from a design ξ and an observation y is the difference between the entropy of the prior and that of the posterior

$$\text{IG}_z(\xi, y) = H[p(z)] - H[p(z \mid y, \xi)]. \quad (1.99)$$

(2023). Differentiable multi-target causal Bayesian experimental design. In *International Conference on Machine Learning* (pp. 34263-34279). However, this work was not included in this thesis as the personal contribution of DRI to the paper was low.

Conditional posterior entropy formulation. We turn $\text{IG}_z(\xi, y)$ into a valid selection criterion by integrating over all possible outcomes y under the design ξ , which has a distribution $p(y | \xi) = \int p(\theta)p(y | \theta, \xi)d\theta = \mathbb{E}_{p(\theta)}[p(y | \theta, \xi)]$ gives us

$$\text{EIG}_z(\xi) = \mathbb{E}_{p(y|\xi)}[\text{IG}_z(\xi, y)] \quad (1.100)$$

$$= H[Z] - H[Z | Y, \xi] = \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{p(z | y, \xi)}{p(z)} \right] \quad (\text{cf. Equation (1.46)}) \quad (1.101)$$

$$= I(Z; Y | \xi) \quad (\text{cf. Equation (1.45)}). \quad (1.102)$$

Expanding the densities inside the log using the expressions we derived in the previous section:

$$\text{EIG}_z(\xi) = \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{p(z | y, \xi)}{p(z)} \right] \quad (1.103)$$

$$= \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{\int p(z | \theta, y, \xi)p(\theta | y, \xi)d\theta}{\mathbb{E}_{p(\theta)}[p(z | \theta)]} \right], \quad p(z | \theta, y, \xi) = p(z | \theta) \quad (1.104)$$

$$= \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{\mathbb{E}_{p(\theta|y,\xi)}[p(z | \theta)]}{\mathbb{E}_{p(\theta)}[p(z | \theta)]} \right]. \quad (1.105)$$

Therefore, to approximate this form of the EIG we need:

1. Samples from the **joint** $y, z \sim p(y, z | \xi)$ for the outer expectation. We do this by sampling $\theta' \sim p(\theta)$, $z \sim p(z | \theta')$, $y \sim p(y | \theta', \xi)$, keeping y, z and ignoring θ' samples. The difficulty of sampling from this joint distribution depends on the specific form of f_z .
2. Samples from the **posterior** given outcomes $\theta \sim p(\theta | y, \xi)$ for the inner expectation in the numerator. Note that if this posterior is intractable this procedure can quickly become computationally prohibitive. For example, if we use M samples to approximate the outer expectation, we need to run M separate MCMC chains, one for each y sampled in Step 1.
3. Samples from the **prior** $\theta \sim p(\theta)$ for the inner expectation in the denominator. This is usually straightforward, as it does not depend on the observed data or the goal-oriented quantity of interest.
4. Being able to evaluate $p(z | \theta)$. The difficulty of this step depends on the specific form of the function f_z . If f_z is deterministic, i.e., there is a one-to-one mapping between θ and z , then evaluating $p(z | \theta)$ is straightforward, whilst if f_z is stochastic or involves complex, non-linear transformations, evaluating $p(z | \theta)$ can be challenging.

Conditional likelihood entropy formulation. Using the properties of mutual information, we can equivalently write

$$\text{EIG}_z(\xi) = H[Y | \xi] - H[Y | Z, \xi] = \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{p(y|z, \xi)}{p(y|z)} \right] \quad (\text{cf. Equation (1.47)}), \quad (1.106)$$

which we can re-arrange and expand as follows:

$$\text{EIG}_z(\xi) = \mathbb{E}_{p(y,z|\xi)} \left[\log \frac{\mathbb{E}_{p(\theta|z)}[p(y|\theta, \xi)]}{\mathbb{E}_{p(\theta)}[p(y|\theta, \xi)]} \right]. \quad (1.107)$$

To approximate this form of the EIG we need

1. Samples from the **joint** $y, z \sim p(y, z | \xi)$ for the outer expectation. We do this by sampling $\theta' \sim p(\theta)$, $z \sim p(z | \theta')$, $y \sim p(y | \theta', \xi)$, keeping y, z and ignoring θ' samples. The difficulty of sampling from this joint distribution depends on the specific form of f_z .
2. Samples from the **posterior** on θ given the QoIs $\theta \sim p(\theta | z)$ for the inner expectation in the numerator. Depending on the relationship between θ and z , this might be easy or not be straightforward. Note that this needs to be done for all z -s sampled in Step 1.
3. Samples from the **prior** $\theta \sim p(\theta)$ for the inner expectation in the denominator. This is usually straightforward, as it does not depend on the observed data or the QoI z .
4. Being able to evaluate $p(y | \theta, \xi)$, which is easy for explicit likelihood-based models.

If these four criteria are met, constructing a nested Monte Carlo estimator for the EIG is straightforward. This situation, where the likelihood function is available but the posterior distribution of the model parameters given the goal-oriented quantity of interest is intractable, is referred to as a *semi-implicit* likelihood model in Foster (2021).

Black-box variational estimators for goal-oriented BED

We can employ black-box variational estimators can be employed to approximate the goal-oriented EIG_z by learning a critic function that approximates the unknown density ratio $\frac{p(z|y, \xi)}{p(z)} = \frac{p(y|z, \xi)}{p(y|z)}$. To use these variational estimators, we only need an efficient way to obtain joint samples $y, z \sim p(y, z | \xi)$, which may be straightforward or challenging depending on the specific problem setting. Once we have these joint samples, we can use estimators such as InfoNCE (1.72), NWJ (1.74), or LF-ACE (1.70) if an analytic prior is available.

1.5 Related fields: Bayesian optimisation, Bayesian active learning and Bayesian Reinforcement Learning

Bayesian experimental design shares many similarities with the related fields of Bayesian optimisation (BO), Bayesian active learning (BAL), and model-based Bayesian reinforcement learning (BLR). While these approaches share the common goal of efficient information gathering and decision-making under uncertainty, they differ in their specific objectives, applications, and the role of the underlying models, as summarised in Table 1.2.

1.5.1 Bayesian optimisation

Bayesian optimisation (BO) (Brochu et al., 2010; Snoek et al., 2012; Shahriari et al., 2015) provides a framework for finding the global maximiser of a black-box objective function, i.e., a function that is unknown in closed form and has unknown gradients. It has been applied in various practical settings, including recommended systems and marketing, where we wish to optimising advertising budgets to maximise returns (Letham and Bakshy, 2019), hyper-parameter optimisation (HPO) and neural architecture search (NAS) (Wu et al., 2019; Ru et al., 2020; Turner et al., 2021; Victoria and Maragatham, 2021), as well as many scientific applications in chemistry (Guo et al., 2023), pharmacology (Sano et al., 2020), fusion (Chung et al., 2020), to name a few.

The objective in BO is to find $\xi^* = \arg \max_{\xi \in \Xi} f(\xi)$. To guide the search, the black-box function f is modelled probabilistically, typically using a Gaussian process prior (Williams and Rasmussen, 2006), although other models can also be considered (Bergstra et al., 2011; Hutter et al., 2013; Shah et al., 2013; Neiswanger et al., 2019).

After the surrogate model has been selected, the next key step is choosing an acquisition function a , that ranks .. Many such functions have been proposed and used, including probability of improvement (PI, Kushner, 1964), expected improvement (EI, Mockus et al., 1978), Thompson sampling (Thompson, 1933), among others. More recently, there has been a focus on more principled, information-theoretic acquisition functions, which choose sampling locations based on the entropy of the distribution of the unknown maximiser ξ^* . Examples include: entropy search (ES, Hennig and Schuler, 2012), predictive entropy search (PES, Hernández-Lobato et al., 2014), max-value entropy search (MES, Wang and Jegelka, 2017), as well as more recent modifications improve their speed, scalability and efficiency (e.g. Alvi et al., 2019; Hvarfner et al., 2022; Nguyen et al., 2022; Ru et al., 2018, among other works).

Overview of key information-theoretic BO objectives

Given data h_t , typical information-theoretic acquisition functions maximise the mutual information between the location of the global optimum $\xi^* = \arg \max_{\xi \in \Xi} f(\xi)$ and the queried point (ξ, y) . Depending on the form chosen, we get ES (Hennig and Schuler, 2012) or PES (Hernández-Lobato et al., 2014):

$$a(\xi) = I(\xi^*; (\xi, y) | h_t) \quad (1.108)$$

$$= H[p(\xi^* | h_t)] - \mathbb{E}_{p(y | h_t, x)}[H[p(\xi^* | (\xi, y) \cup h_t)]] \quad \text{ES, cf. Equation (1.46)} \quad (1.109)$$

$$= H[p(y | h_t, \xi)] - \mathbb{E}_{p(\xi^* | h_t)}[H[p(y | h_t, x, \xi^*)]] \quad \text{PES, cf. Equation (1.47)} \quad (1.110)$$

Instead, MES (Wang and Jegelka, 2017) considers the mutual information between the queried point (ξ, y) and the maximum value, $y^* = f(\xi^*)$, instead of its argmax:

$$a^{\text{MES}}(\xi) = I(y^*; (\xi, y) | h_t) = H[p(y | h_t, \xi)] - \mathbb{E}_{p(\xi^* | h_t)}[H[p(y | h_t, x, y^*)]] \quad (1.111)$$

Remark 1.36. *BO with information based objective can be viewed as a special case of the goal-oriented BED framework discussed in §1.4. In this context, the goal or quantity of interest in BO is either the maximum value of the objective function y^* (in MES) or the the global maximiser ξ^* (in ES or PES). See Example 1.34 for more details.*

Computing the mutual information acquisition function. As discussed, mutual information (MI) is doubly intractable to compute in general. In BO, it is common to choose a GP as the surrogate model due to their flexibility and tractability. When working with GPs we can use closed-form expressions or efficient approximations to compute the MI acquisition effectively.

If we choose to employ a more general model, any of the estimators previously discussed can be used (see Table 1.1 for an overview). For example, in Chapter §5 we combine max-value entropy search (MES) with the InfoNCE estimator for contextual optimisation.

1.5.2 Bayesian active learning

Bayesian active learning (BAL) is another closely related field to BED. The aim of BAL is to train machine learning models in a data-efficient manner by iteratively acquiring the most informative labels (MacKay, 1992; Cohn et al., 1996). BAL can be viewed as a special case of BED (Rainforth et al., 2024; Foster, 2021), where the designs correspond to the inputs for which

labels are acquired, typically chosen from a finite pool of unlabeled data points.

In the context of BAL, the EIG for the model parameters is often referred to as the Bayesian Active Learning by Disagreement (BALD) score (Houlsby et al., 2011). It is exactly equivalent to the EIG as presented in Equation (1.47) (see Remark 1.22), and quantifies the expected reduction in uncertainty about the model parameters after acquiring the label for a particular data point. One of the key challenges in BAL is performing active learning in batches, which involves combinatorial optimisation over the pool of unlabeled data points. Various BAL methods have been proposed in the literature to address this challenge, such as BADGE (Ash et al., 2019), BatchBALD (Kirsch et al., 2019), and PowerBALD and its variants (Kirsch et al., 2021).

For a comprehensive overview of BAL, its applications, and recent advances, please refer to Ren et al. (2021), Zhan et al. (2022), or Kirsch (2024).

1.5.3 Bayesian model-based reinforcement learning

Reinforcement learning (RL, Sutton, 1990; Sutton and Barto, 2018; Arulkumaran et al., 2017), including Bayesian formulations (Ghavamzadeh et al., 2015), is a vast subfield of machine learning, broadly focused on optimal decision-making. More concretely, in an RL setup, an agent interacts with its environment (simulated or real), governed by a Markov Decision Process (MDP), that provides a formal framework for modeling the state transitions (environment dynamics) and rewards (or penalties) based on the agent’s actions. The goal is to find an action-selection strategy, or *policy*, that optimises long-term reward. This process involves balancing exploration (trying new actions) and exploitation (using actions that are known to give high rewards). This trade-off is particularly important in scenarios with sparse rewards (Ladosz et al., 2022).

Considering the broad scope and generality of the RL framework, it is not surprising that many problems, including BED, can be formulated within it. Indeed, Foster (2021) showed that the sequential experimental design problem described in Section §1.3.3 can be treated as a special case of a Bayes-adaptive Markov Decision Process (BAMDP, Ross et al., 2007; Guez et al., 2012). In this formulation, the transition model of the BAMDP corresponds to the model parameters θ , whilst the reward is defined by the incremental EIG. Unlike conventional RL rewards, such as scoring points in a game, the EIG functions as an *intrinsic reward* promoting knowledge acquisition about the model environment.

We can therefore view adaptive BED as making sequential decisions with the goal of learning about the underlying model parameters, driven by an intrinsic, information-based reward (namely, mutual information), which is based on the uncertainty the agent has about the envi-

ronment. If the agent is exploring highly uncertain areas of the environment, the reward is high. Various methods that utilise different approximation techniques for computing this mutual information reward have been developed (Kolter and Ng, 2009; Still and Precup, 2012; Mohamed and Rezende, 2015; Houthoofd et al., 2016). It is worth noting that in RL, other types of intrinsic rewards are also possible, such as rewarding novel behaviours (Lehman and Stanley, 2011; Risi et al., 2009; Hong et al., 2018).

This view of BED falls into the subfield of RL known as pure (or unsupervised) exploration (see Ladosz et al., 2022, for a review), more concretely within intrinsically motivated model-based reinforcement learning (Singh et al., 2005). However, a key distinction exists between conventional BED and conventional RL approaches: in BED, the models are often mechanistic and provided by scientists and domain experts. In contrast, in RL, the dynamics models are typically learned through interactions with the environment, or from a static dataset of state-action-reward trajectories (offline RL, Agarwal et al., 2020; Levine et al., 2020)).

Remark 1.37 (BED and System Identification). ***System Identification** in robotics (Ljung, 2010; Nelles and Nelles, 2020; Zhao et al., 2020; Höfer et al., 2021) can be viewed as a sub-field or special case of model-based RL, where the goal is to identify a dynamics model from data, as opposed to merely fitting any model to data, which is more common in conventional RL settings. BED can also be viewed as a system identification problem, as it aims to efficiently learn about the underlying system dynamics through strategic experimentation.*

Remark 1.38 (BED and Active Inference). ***Active Inference** in neuroscience (Friston et al., 2016, 2017; Smith et al., 2022) shares core conceptual similarities with both model-based RL and BED. The active inference framework posits that perception and learning are processes of minimising variational free energy, while action selection, planning, and decision-making are understood in terms of minimising expected free energy (Friston, 2010). Free energy minimisation is closely linked to information maximisation, Bayesian inference, and Bayesian model selection. Despite these similarities, active inference is a distinct framework with its own set of assumptions and implications. This opens avenues for future research to explore connections between the two fields, potentially enriching both by integrating insights and methodologies.*

Remark 1.39. *Finally, we note that the relationship between BED and exploration in model-based RL becomes even more apparent in the context of the policy-based approach to BED (PB-BED) that we introduce in this thesis (§2, §3, §4). Future work may further elucidate connections, exploring how these methodologies can be integrated to enhance both the theoretical understanding and practical applications of PB-BED.*

	BED	Bayesian optimisation	Bayesian active learning	Bayesian RL
Objective	System identification	Black-box function optimisation	Data-efficient supervised learning	Reward maximisation
Achieved by	Optimising experimental designs to maximize information gain	Iteratively selecting maximally informative query points	Selecting the most informative (batch of) data points to label	Learning a policy (function) that maximises reward
Model	Parametric, mechanistic, given	Nonparametric	Nonparametric or parametric, learnt	Nonparametric or parametric, learnt
Role of the model	Represent relationship between designs and outcomes	Acts as a surrogate to approximate the expensive objective	Represent relationship between input features and labels	Represent the dynamics of the environment (world model)
Acquisition	Sequential or batch	Sequential	Sequential or batch	Sequential
Output	Updated probabilistic model (posterior on model parameters)	Global optimum of the function (not the function itself)	Updated probabilistic model (posterior on model parameters)	Optimal policy
Applications	Scientific experiments, clinical trials, survey design	HPO, NAS, recommender systems, scientific experiments	Classification, regression with limited labels	Autonomous systems, game playing, robotics

Table 1.2: **Comparing Bayesian Experimental Design (BED), Bayesian Optimization (BO), Bayesian Active Learning (BAL), and Model-Based Bayesian Reinforcement Learning (BLR)**. Whilst these approaches share the common high-level goal of efficient information gathering and decision-making under uncertainty, they differ in their specific objectives, applications, and the role of the underlying models. BED focuses on system identification, BO on black-box function optimisation, BAL on data-efficient supervised learning, and BRL on reward maximization. Note that we present the **most common and typical cases for each approach, highlighting that this is not an exhaustive list** and that there is significant variation and overlap between them. The specific formulation and application of each approach may vary depending on the problem setting and the assumptions made about the underlying models. All four frameworks can be viewed as different manifestations of the same underlying principles of Bayesian decision-making and information theory, and the boundaries between them are often blurred in practice. Despite these variations, the comparison in the table provides a high-level overview of the key differences and similarities between these approaches.

Chapter references

- Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. {TensorFlow}: a system for {Large-Scale} machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pages 265–283, 2016. 41
- Lorenz Adlung, Yotam Cohen, Uria Mor, and Eran Elinav. Machine learning in clinical decision making. *Med*, 2(6):642–665, 2021. 7
- Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An optimistic perspective on offline reinforcement learning. In *International Conference on Machine Learning*, pages 104–114. PMLR, 2020. 55
- Ahsan S Alvi, Binxin Ru, Jan Calliess, Stephen J Roberts, and Michael A Osborne. Asynchronous batch bayesian optimisation with improved local penalisation. *arXiv preprint arXiv:1901.10452*, 2019. 52
- Panagiotis Annadani, Yashas anf Tigas, Desi R Ivanova, Andrew Jesson, Yarin Gal, Adam Foster, and Stefan Bauer. Differentiable Multi-Target Causal Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 34263–34279. PMLR, 2023. 48
- Theo Araujo, Natali Helberger, Sanne Kruikemeier, and Claes H De Vreese. In ai we trust? perceptions about automated decision-making by artificial intelligence. *AI & society*, 35: 611–623, 2020. 7
- Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Guided image generation with conditional invertible neural networks. *arXiv preprint arXiv:1907.02392*, 2019. 39
- Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017. 54
- Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019. 54
- AC Atkinson and VV Fedorov. The design of experiments for discriminating between two rival models. *Biometrika*, 62(1):57–70, 1975. 8

- Ahmed Attia, Alen Alexanderian, and Arvind K Saibaba. Goal-oriented optimal design of experiments for large-scale bayesian linear inverse problems. *Inverse Problems*, 34(9):095009, 2018. 47
- David Barber and Felix Agakov. The IM algorithm: a variational approach to information maximization. *Advances in Neural Information Processing Systems*, 16:201–208, 2003. 30
- Solon Barocas and Andrew D Selbst. Big data’s disparate impact. *Calif. L. Rev.*, 104:671, 2016. 7
- Joakim Beck, Ben Mansour Dia, Luis FR Espath, Quan Long, and Raul Tempone. Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain. *Computer Methods in Applied Mechanics and Engineering*, 334:523–553, 2018. 28, 29, 32
- Ishmael Belghazi, Sai Rajeswar, Aristide Baratin, R Devon Hjelm, and Aaron Courville. MINE: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018. 35
- James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyperparameter optimization. *Advances in neural information processing systems*, 24, 2011. 52
- José M Bernardo and Adrian FM Smith. *Bayesian theory*, volume 405. John Wiley & Sons, 2009. 11
- Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017. 13
- Tom Blau, Edwin V Bonilla, Iadine Chades, and Amir Dezfouli. Optimizing sequential experimental design with deep reinforcement learning. In *International Conference on Machine Learning*, pages 2107–2128. PMLR, 2022. 37
- David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017. 13
- Eric Bogert, Aaron Schecter, and Richard T Watson. Humans rely more on algorithms than social influence as a task becomes more difficult. *Scientific reports*, 11(1):8028, 2021. 7
- George EP Box. Choice of response surface design and alphabetic optimality. Technical report, Wisconsin University-Madison Mathematics Research Center, 1982. 8
- Joan Fisher Box. Ra fisher and the design of experiments, 1922–1926. *The American Statistician*, 34(1):1–7, 1980. 8

- Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010. 52
- Lukas Budach, Moritz Feuerpfeil, Nina Ihde, Andrea Nathansen, Nele Noack, Hendrik Patzlaff, Felix Naumann, and Hazar Harmouch. The effects of data quality on machine learning performance. *arXiv preprint arXiv:2207.14529*, 2022. 7
- Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995. 8, 9, 10
- Youngseog Chung, Ian Char, Willie Neiswanger, Kirthevasan Kandasamy, Andrew Oakleigh Nelson, Mark D Boyer, Egemen Kolenen, and Jeff Schneider. Offline contextual bayesian optimization for nuclear fusion. *arXiv preprint arXiv:2001.01793*, 2020. 52
- David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996. 53
- Alex R Cook, Gavin J Gibson, and Christopher A Gilligan. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008. 9, 15, 28
- Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806, 2017. 7
- Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 1999. 16
- Mahasen B. Dehideniya, Christopher C. Drovandi, and James M. McGree. Optimal bayesian design for discriminating between models with intractable likelihoods in epidemiology. *Computational Statistics & Data Analysis*, 124:277–297, 2018. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2018.03.004>. 46
- Persi Diaconis and Donald Ylvisaker. Conjugate priors for exponential families. *The Annals of statistics*, pages 269–281, 1979. 12
- Gustav Elfving. Optimum allocation in linear regression theory. *The Annals of Mathematical Statistics*, pages 255–262, 1952. 8
- Ronald Aylmer Fisher. *Design of experiments*. 1935. 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. 9, 29, 30, 31, 32, 37
- 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. 9, 29, 32, 33, 34, 37
- Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing Sequential Bayesian Experimental Design. In *International conference on machine learning*, pages 3384–3395. PMLR, 2021. 37
- Adam Evan Foster. *Variational, Monte Carlo and policy-based approaches to Bayesian experimental design*. PhD thesis, University of Oxford, 2021. 25, 51, 53, 54
- Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010. 55
- Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, Giovanni Pezzulo, et al. Active inference and learning. *Neuroscience & Biobehavioral Reviews*, 68:862–879, 2016. 55
- Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active inference: a process theory. *Neural computation*, 29(1):1–49, 2017. 55
- Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013. 11
- Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian workflow. *arXiv preprint arXiv:2011.01808*, 2020. 11
- Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6): 721–741, 1984. 13
- Samuel Gershman and Noah Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014. 38

- Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, Aviv Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015. 54
- Jinwoo Go and Tobin Isaac. Robust expected information gain for optimal bayesian experimental design using ambiguity sets. *38th Conference on Uncertainty in Artificial Intelligence*, 2022. 9
- Javier González, Michael Osborne, and Neil Lawrence. Glasses: Relieving the myopia of bayesian optimisation. In *Artificial Intelligence and Statistics*, pages 790–799. PMLR, 2016. 44
- Arthur Guez, David Silver, and Peter Dayan. Efficient Bayes-adaptive reinforcement learning using sample-based search. In *Advances in neural information processing systems*, pages 1025–1033, 2012. 54
- Jeff Guo, Bojana Ranković, and Philippe Schwaller. Bayesian optimization for chemical reactions. *Chimia*, 77(1/2):31–38, 2023. 52
- M Hamada, HF Martz, CS Reese, and AG Wilson. Finding near-optimal bayesian experimental designs via genetic algorithms. *The American Statistician*, 55(3):175–181, 2001. 28, 37
- Cong Han and Kathryn Chaloner. Bayesian experimental design for nonlinear mixed-effects models with application to hiv dynamics. *Biometrics*, 60(1):25–33, 2004. 9, 37
- W Keith Hastings. *Monte Carlo sampling methods using Markov chains and their applications*. Oxford University Press, 1970. 13
- Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(Jun):1809–1837, 2012. 52, 53
- José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in neural information processing systems*, pages 918–926, 2014. 48, 52, 53
- Sebastian Höfer, Kostas Bekris, Ankur Handa, Juan Camilo Gamboa, Melissa Mozifian, Florian Golemo, Chris Atkeson, Dieter Fox, Ken Goldberg, John Leonard, et al. Sim2real in robotics and automation: Applications and challenges. *IEEE transactions on automation science and engineering*, 18(2):398–400, 2021. 55
- Zhang-Wei Hong, Tzu-Yun Shann, Shih-Yang Su, Yi-Hsiang Chang, Tsu-Jui Fu, and Chun-Yi Lee. Diversity-driven exploration strategy for deep reinforcement learning. *Advances in neural information processing systems*, 31, 2018. 55

- Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011. 25, 54
- Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *arXiv preprint arXiv:1605.09674*, 2016. 55
- Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013. 29
- Xun Huan and Youssef M Marzouk. Sequential bayesian optimal experimental design via approximate dynamic programming. *arXiv preprint arXiv:1604.08320*, 2016. 43
- Ben Hutchinson, Andrew Smart, Alex Hanna, Emily Denton, Christina Greer, Oddur Kjartansson, Parker Barnes, and Margaret Mitchell. Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 560–575, 2021. 7
- Frank Hutter, Holger Hoos, and Kevin Leyton-Brown. An evaluation of sequential model-based optimization for expensive blackbox functions. In *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, pages 1209–1216, 2013. 52
- Carl Hvarfner, Frank Hutter, and Luigi Nardi. Joint entropy search for maximally-informed bayesian optimization. *Advances in Neural Information Processing Systems*, 35:11494–11506, 2022. 52
- Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael U Gutmann, and Thomas Rainforth. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. *Advances in Neural Information Processing Systems*, 34, 2021. 37
- Desi R Ivanova, Joel Jennings, Tom Rainforth, Cheng Zhang, and Adam Foster. CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 14445–14464. PMLR, 2023. 48
- Shali Jiang, Henry Chai, Javier Gonzalez, and Roman Garnett. Binoculars for efficient, nonmyopic sequential experimental design. In *International Conference on Machine Learning*, pages 4794–4803. PMLR, 2020. 44
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 40
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *ICLR*, 2014. 38, 40

- Andreas Kirsch. Advancing deep active learning & data subset selection: Unifying principles with information-theory intuitions. *arXiv preprint arXiv:2401.04305*, 2024. 54
- Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32:7026–7037, 2019. 54
- Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal. Stochastic batch acquisition: A simple baseline for deep active learning. *arXiv preprint arXiv:2106.12059*, 2021. 54
- Steven Kleinegesse and Michael Gutmann. Efficient Bayesian experimental design for implicit models. *arXiv preprint arXiv:1810.09912*, 2018. 37, 46
- 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. URL <http://proceedings.mlr.press/v119/kleinegesse20a.html>. 9, 29, 35, 37
- Steven Kleinegesse and Michael U. Gutmann. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds, 2021. 9, 29
- Steven Kleinegesse, Christopher Drovandi, and Michael U Gutmann. Sequential bayesian experimental design for implicit models via mutual information. *arXiv preprint arXiv:2003.09379*, 2020. 46
- J Zico Kolter and Andrew Y Ng. Near-bayesian exploration in polynomial time. In *Proceedings of the 26th annual international conference on machine learning*, pages 513–520, 2009. 55
- Harold J Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *J of Basic Engineering*, 1964. 52
- Pawel Ladosz, Lilian Weng, Minwoo Kim, and Hyondong Oh. Exploration in deep reinforcement learning: A survey. *Information Fusion*, 85:1–22, 2022. 54, 55
- Joel Lehman and Kenneth O Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2):189–223, 2011. 55
- Benjamin Letham and Eytan Bakshy. Bayesian optimization for policy search via online-offline experimentation. *Journal of Machine Learning Research*, 20(145):1–30, 2019. 52

- Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020. 55
- Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956. 8, 10, 23
- Dennis V Lindley. *Bayesian statistics, a review*, volume 2. SIAM, 1972. 9, 25
- Lennart Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010. 55
- Quan Long, Marco Scavino, Raúl Tempone, and Suojin Wang. Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations. *Computer Methods in Applied Mechanics and Engineering*, 259:24–39, 2013. 28
- Quan Long, Mohammad Motamed, and Raúl Tempone. Fast bayesian optimal experimental design for seismic source inversion. *Computer Methods in Applied Mechanics and Engineering*, 291:123–145, 2015. 28
- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 40
- David JC MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992. 53
- David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003. 16
- Tatsuya Matsushima, Hiroki Furuta, Yutaka Matsuo, Ofir Nachum, and Shixiang Gu. Deployment-efficient reinforcement learning via model-based offline optimization. *arXiv preprint arXiv:2006.03647*, 2020. 42
- Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021. 7
- Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953. 13

- Petrus Mikkola, Osvaldo A Martin, Suyog Chandramouli, Marcelo Hartmann, Oriol Abril Pla, Owen Thomas, Henri Pesonen, Jukka Corander, Aki Vehtari, Samuel Kaski, et al. Prior knowledge elicitation: The past, present, and future. *arXiv preprint arXiv:2112.01380*, 2112, 2021. 11
- Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of bayesian methods for seeking the extremum. *Towards global optimization*, 2(117-129):2, 1978. 52
- Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. *arXiv preprint arXiv:1509.08731*, 2015. 40, 55
- Peter Müller, Don A Berry, Andrew P Grieve, and Michael Krams. A bayesian decision-theoretic dose-finding trial. *Decision analysis*, 3(4):197–207, 2006. 9
- Willie Neiswanger, Kirthevasan Kandasamy, Barnabas Poczos, Jeff Schneider, and Eric Xing. Probo: Versatile bayesian optimization using any probabilistic programming language. *arXiv preprint arXiv:1901.11515*, 2019. 52
- Oliver Nelles and Oliver Nelles. *Nonlinear dynamic system identification*. Springer, 2020. 55
- Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Rectified max-value entropy search for bayesian optimization. *arXiv preprint arXiv:2202.13597*, 2022. 52
- 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. 35
- Eirini Ntoutsi, Pavlos Fafalios, Ujwal Gadiraju, Vasileios Iosifidis, Wolfgang Nejdl, Maria-Esther Vidal, Salvatore Ruggieri, Franco Turini, Symeon Papadopoulos, Emmanouil Krasanakis, et al. Bias in data-driven artificial intelligence systems—an introductory survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3):e1356, 2020. 7
- 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. 37
- Anthony O’Hagan. Expert knowledge elicitation: subjective but scientific. *The American Statistician*, 73(sup1):69–81, 2019. 11
- Andrei Paleyes, Raoul-Gabriel Urma, and Neil D Lawrence. Challenges in deploying machine learning: a survey of case studies. *ACM computing surveys*, 55(6):1–29, 2022. 7

- Costas Papadimitriou. Optimal sensor placement methodology for parametric identification of structural systems. *Journal of sound and vibration*, 278(4-5):923–947, 2004. 28
- George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021. 39
- 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. 41
- Amandalynne Paullada, Inioluwa Deborah Raji, Emily M Bender, Emily Denton, and Alex Hanna. Data and its (dis) contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11), 2021. 7
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017. 48
- Ben Poole, Sherjil Ozair, Aäron van den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180, 2019. 35
- 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. 37
- Maria Priestley, Fionntán O’donnell, and Elena Simperl. A survey of data quality requirements that matter in ml development pipelines. *ACM Journal of Data and Information Quality*, 15(2):1–39, 2023. 7
- Stefan T Radev, Ulf K Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Köthe. Bayesflow: Learning complex stochastic models with invertible neural networks. *IEEE transactions on neural networks and learning systems*, 33(4):1452–1466, 2020. 39
- Stefan T Radev, Marvin Schmitt, Valentin Pratz, Umberto Picchini, Ullrich Köthe, and Paul-Christian Bürkner. Jana: Jointly amortized neural approximation of complex bayesian models. In *Uncertainty in Artificial Intelligence*, pages 1695–1706. PMLR, 2023. 39

- Howard Raiffa and Robert Schlaifer. Applied statistical decision theory. 1961. 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. 27, 28
- Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern Bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024. 8, 10, 25, 53
- Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021. 54
- Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015. 39
- 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. 40
- Sebastian Risi, Sandy D Vanderbleek, Charles E Hughes, and Kenneth O Stanley. How novelty search escapes the deceptive trap of learning to learn. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 153–160, 2009. 55
- Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951. 39
- Stephane Ross, Brahim Chaib-draa, and Joelle Pineau. Bayes-adaptive pomdps. In *NIPS*, pages 1225–1232, 2007. 54
- Binxin Ru, Michael A Osborne, Mark McLeod, and Diego Granziol. Fast information-theoretic bayesian optimisation. In *International Conference on Machine Learning*, pages 4384–4392. PMLR, 2018. 52
- Binxin Ru, Xingchen Wan, Xiaowen Dong, and Michael Osborne. Interpretable neural architecture search via bayesian optimisation with weisfeiler-lehman kernels. *arXiv preprint arXiv:2006.07556*, 2020. 52
- Elizabeth G. Ryan, Christopher C. Drovandi, M. Helen Thompson, and Anthony N. Pettitt. Towards bayesian experimental design for nonlinear models that require a large number of

- sampling times. *Computational Statistics & Data Analysis*, 70:45–60, 2014. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2013.08.017>. URL <https://www.sciencedirect.com/science/article/pii/S0167947313003149>. 28
- 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. 8, 9, 10
- Kenneth J Ryan. Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *Journal of Computational and Graphical Statistics*, 12(3):585–603, 2003. 37
- Thomas P Ryan and JP Morgan. Modern experimental design. *Journal of Statistical Theory and Practice*, 1(3-4):501–506, 2007. 8
- Syusuke Sano, Tadashi Kadowaki, Koji Tsuda, and Susumu Kimura. Application of bayesian optimization for pharmaceutical product development. *Journal of Pharmaceutical Innovation*, 15:333–343, 2020. 52
- Amar Shah, Andrew G Wilson, and Zoubin Ghahramani. Bayesian optimization using student-t processes. In *NIPS Workshop on Bayesian Optimization*, 2013. 52
- Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015. 52
- Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948. 8, 10
- Satinder Singh, Andrew G Barto, and Nuttapon Chentanez. Intrinsically motivated reinforcement learning. Technical report, MASSACHUSETTS UNIV AMHERST DEPT OF COMPUTER SCIENCE, 2005. 55
- Kirstine Smith. On the standard deviations of adjusted and interpolated values of an observed polynomial function and its constants and the guidance they give towards a proper choice of the distribution of observations. *Biometrika*, 12(1/2):1–85, 1918. 8
- Ryan Smith, Karl J Friston, and Christopher J Whyte. A step-by-step tutorial on active inference and its application to empirical data. *Journal of mathematical psychology*, 107:102632, 2022. 55

- Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012. 47, 52
- Susanne Still and Doina Precup. An information-theoretic approach to curiosity-driven reinforcement learning. *Theory in Biosciences*, 131:139–148, 2012. 55
- Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, pages 216–224. Elsevier, 1990. 54
- Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018. 54
- William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933. 52
- Panagiotis Tigas, Yashas Annadani, Andrew Jesson, Bernhard Schölkopf, Yarin Gal, and Stefan Bauer. Interventions, where and how? experimental design for causal models at scale. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 48
- Christian Toth, Lars Lorch, Christian Knoll, Andreas Krause, Franz Pernkopf, Robert Peharz, and Julius Von Kügelgen. Active bayesian causal inference. *Advances in Neural Information Processing Systems*, 35:16261–16275, 2022. 48
- Ryan Turner, David Eriksson, Michael McCourt, Juha Kiili, Eero Laaksonen, Zhen Xu, and Isabelle Guyon. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In *NeurIPS 2020 Competition and Demonstration Track*, pages 3–26. PMLR, 2021. 52
- Jojanneke van den Berg, Andrew Curtis, and Jeannot Trampert. Optimal nonlinear bayesian experimental design: an application to amplitude versus offset experiments. *Geophysical Journal International*, 155(2):411–421, 2003. 37
- Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 34, 35
- A Helen Victoria and Ganesh Maragatham. Automatic tuning of hyperparameters using bayesian optimization. *Evolving Systems*, 12(1):217–223, 2021. 52

- Julius von Kügelgen, Paul K Rubenstein, Bernhard Schölkopf, and Adrian Weller. Optimal experimental design via bayesian optimization: active causal structure learning for gaussian process networks. *arXiv preprint arXiv:1910.03962*, 2019. 37
- Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008. 13
- Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient Bayesian optimization. In *International Conference on Machine Learning*, pages 3627–3635. PMLR, 2017. 48, 52, 53
- Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006. 52
- Christina Winkler, Daniel E Worrall, Emiel Hoogeboom, and Max Welling. Learning likelihoods with conditional normalizing flows. 2019. 39
- Robert L Winkler. The assessment of prior distributions in bayesian analysis. *Journal of the American Statistical association*, 62(319):776–800, 1967. 11
- Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17(1):26–40, 2019. 52
- Keyi Wu, Peng Chen, and Omar Ghattas. An efficient method for goal-oriented linear bayesian optimal experimental design: Application to optimal sensor placemen. *arXiv preprint arXiv:2102.06627*, 2021. 47
- Andrea Zanette, Kefan Dong, Jonathan N Lee, and Emma Brunskill. Design of experiments for stochastic contextual linear bandits. *Advances in Neural Information Processing Systems*, 34: 22720–22731, 2021. 42
- Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022. 54
- 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. 13
- Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, pages 737–744. IEEE, 2020. 55

Sue Zheng, Jason Pacheco, and John Fisher. A robust approach to sequential information theoretic planning. In *International Conference on Machine Learning*, pages 5941–5949, 2018. 27

Shijie Zhong, Wanggang Shen, Tommie Catanach, and Xun Huan. Goal-oriented bayesian optimal experimental design for nonlinear models using markov chain monte carlo. *arXiv preprint arXiv:2403.18072*, 2024. 47

Chapter 2

Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design

This paper is published as:

Adam Foster*, Desi R Ivanova*, Ilyas Malik, and Thomas Rainforth. Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design. In *Proceedings of the 38th International Conference on Machine Learning*, pages 3384–3395. PMLR, 2021.

*Equal contribution

Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design

Adam Foster^{* 1} Desi R. Ivanova^{* 1} Ilyas Malik² Tom Rainforth¹

Abstract

We introduce *Deep Adaptive Design* (DAD), a method for amortizing the cost of adaptive Bayesian experimental design that allows experiments to be run in real-time. Traditional sequential Bayesian optimal experimental design approaches require substantial computation at *each* stage of the experiment. This makes them unsuitable for most real-world applications, where decisions must typically be made quickly. DAD addresses this restriction by learning an amortized *design network* upfront and then using this to rapidly run (multiple) adaptive experiments at deployment time. This network represents a *design policy* which takes as input the data from previous steps, and outputs the next design using a single forward pass; these design decisions can be made in milliseconds during the live experiment. To train the network, we introduce contrastive information bounds that are suitable objectives for the sequential setting, and propose a customized network architecture that exploits key symmetries. We demonstrate that DAD successfully amortizes the process of experimental design, outperforming alternative strategies on a number of problems.

1. Introduction

A key challenge across disciplines as diverse as psychology (Myung et al., 2013), bioinformatics (Vanlier et al., 2012), pharmacology (Lyu et al., 2019) and physics (Dushenko et al., 2020) is to design experiments so that the outcomes will be as informative as possible about the underlying process. Bayesian optimal experimental design (BOED) is a powerful mathematical framework for tackling this problem (Lindley, 1956; Chaloner & Verdinelli, 1995).

In the BOED framework, outcomes y are modeled in a Bayesian manner (Gelman et al., 2013; Kruschke, 2014)

^{*}Equal contribution ¹Department of Statistics, University of Oxford, UK ²Work undertaken whilst at the University of Oxford. Correspondence to: Adam Foster <adam.foster@stats.ox.ac.uk>.

using a likelihood $p(y|\theta, \xi)$ and a prior $p(\theta)$, where ξ is our controllable design and θ is the set of parameters we wish to learn about. We then optimize ξ to maximize the *expected information gained* about θ (equivalently the mutual information between y and θ):

$$I(\xi) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)} [\log p(y|\theta, \xi) - \log p(y|\xi)]. \quad (1)$$

The true power of BOED is realized when it is used to design a sequence of experiments $\xi_1, \dots, \xi_T$, wherein it allows us to construct *adaptive* strategies which utilize information gathered from past data to tailor each successive design ξ_t during the progress of the experiment. The conventional, iterative, approach for selecting each ξ_t is to fit the posterior $p(\theta|\xi_{1:t-1}, y_{1:t-1})$ representing the updated beliefs about θ after $t-1$ iterations have been conducted, and then substitute this for the prior in (1) (Ryan et al., 2016; Rainforth, 2017; Kleinegesse et al., 2020). The design ξ_t is then chosen as the one which maximizes the resulting objective.

Unfortunately, this approach necessitates significant computational time to be expended *between each step of the experiment* in order to update the posterior and compute the next optimal design. In particular, $I(\xi)$ is doubly intractable (Rainforth et al., 2018; Zheng et al., 2018) and its optimization constitutes a significant computational bottleneck. This can be prohibitive to the practical application of sequential BOED as design decisions usually need to be made quickly for the approach to be useful (Evans & Mathur, 2005).

To give a concrete example, consider running an adaptive survey to understand political opinions (Pasek & Kroshnick, 2010). A question ξ_t is put to a participant who gives their answer y_t and this data is used to update an underlying model with latent variables θ . Here sequential BOED is of immense value because previous answers can be used to guide future questions, ensuring that they are pertinent to the particular participant. However, it is not acceptable to have lengthy delays between questions to compute the next design, precluding existing approaches from being used.

To alleviate this problem, we propose *amortizing* the cost of sequential experimental design, performing upfront training before the start of the experiment to allow very fast design decisions at deployment, when time is at a premium. This amortization is particularly useful in the common scenario where the same adaptive experimental framework will be

deployed numerous times (e.g. having multiple participants in a survey). Here amortization not only removes the computational burden from the live experiment, it also allows for sharing computation across multiple experiments, analogous to inference amortization that allows one to deal with multiple datasets (Stuhlmüller et al., 2013).

Our approach, called **Deep Adaptive Design (DAD)**, constructs a single *design network* which takes as input the designs and observations from previous stages, and outputs the design to use for the next experiment. The network is learned by simulating hypothetical experimental trajectories and then using these to train the network to make near-optimal design decisions automatically. That is, it learns a *design policy* which makes decisions as a function of the past data, and we optimize the parameters of this policy rather than an individual design. Once learned, the network eliminates the computational bottleneck at each iteration of the experiment, enabling it to be run both adaptively and quickly; it can also be used repeatedly for different instantiations of the experiment (e.g. different human participants).

To allow for efficient, effective, and simple training, we show how DAD networks can be learned without any direct posterior or marginal likelihood estimation. This is achieved by reformulating the sequential BOED problem from its conventional iterative form, to a single holistic objective based on the *overall* expected information gained from the entire experiment when using a policy to make each design decision deterministically given previous design outcome pairs. We then derive contrastive bounds on this objective that allow for end-to-end training of the policy parameters with stochastic gradient ascent, thereby sidestepping both the need for inference and the double intractability of the EIG objective. This approach has the further substantial benefit of allowing non-myopic adaptive strategies to be learned, that is strategies which take account of their own future decisions, unlike conventional approaches.

We further demonstrate a key permutation symmetry property of the optimal design policy, and use this to propose a customized architecture for the experimental design network. This is critical to allowing effective amortization across time steps. The overall result of the theoretical formulation, novel contrastive bounds, and neural architecture is a methodology which enables us to bring the power of deep learning to bear on adaptive experimental design.

We apply DAD to a range of problems relevant to applications such as epidemiology, physics and psychology. We find that DAD is able to accurately amortize experiments, opening the door to running adaptive BOED in real time.

2. Background

Because experimentation is a potentially costly endeavour, it is essential to design experiments in manner that maximizes

the amount of information garnered. The BOED framework, pioneered by Lindley (1956), provides a powerful means of doing this in a principled manner. Its key idea is to optimize the experimental design ξ to maximize the expected amount of *information* that will be gained about the latent variables of interest, θ , upon observing the experiment outcome y .

To implement this approach, we begin with the standard Bayesian modelling set-up consisting of an explicit likelihood model $p(y|\theta, \xi)$ for the experiment, and a prior $p(\theta)$ representing our initial beliefs about the unknown latent. After running a hypothetical experiment with design ξ and observing y , our updated beliefs are the posterior $p(\theta|\xi, y)$. The amount of information that has been gained about θ can be mathematically described by the reduction in entropy from the prior to the posterior

$$\text{IG}(\xi, y) = H[p(\theta)] - H[p(\theta|\xi, y)]. \quad (2)$$

The *expected information gain* (EIG) is formed by taking the expectation over possible outcomes y , using the model itself to simulate these. Namely we take an expectation with respect to $y \sim p(y|\xi) = \mathbb{E}_{p(\theta)}[p(y|\theta, \xi)]$, yielding

$$\begin{aligned} I(\xi) &:= \mathbb{E}_{p(y|\xi)} [\text{IG}(\xi, y)] \\ &= \mathbb{E}_{p(\theta)p(y|\theta, \xi)} [\log p(\theta|\xi, y) - \log p(\theta)] \\ &= \mathbb{E}_{p(\theta)p(y|\theta, \xi)} [\log p(y|\theta, \xi) - \log p(y|\xi)] \end{aligned}$$

which is the mutual information between y and θ under design ξ . The optimal design is defined as $\xi^* = \arg \max_{\xi \in \Xi} I(\xi)$, where Ξ is the space of feasible designs.

It is common in BOED settings to be able to run multiple experiment iterations with designs $\xi_1, \dots, \xi_T$, observing respective outcomes $y_1, \dots, y_T$. One simple strategy for this case is *static* design, also called fixed or batch design, which selects all $\xi_1, \dots, \xi_T$ before making any observation. The designs are optimized to maximize the EIG, with $y_{1:T}$ in place of y and $\xi_{1:T}$ in place of ξ , effectively treating the whole sequence of experiments as one experiment with enlarged observation and design spaces.

2.1. Conventional adaptive BOED

This static design approach is generally sub-optimal as it ignores the fact that information from previous iterations can substantially aid in the design decisions at future iterations. The power of the BOED framework can thus be significantly increased by using an *adaptive* design strategy that chooses each ξ_t dependent upon $\xi_{1:t-1}, y_{1:t-1}$. This enables us to use what has already been learned in previous experiments to design the next one optimally, resulting in a virtuous cycle of refining beliefs and using our updated beliefs to design good experiments for future iterations.

The conventional approach to computing designs adaptively is to fit the posterior distribution $p(\theta|\xi_{1:t-1}, y_{1:t-1})$ at each

step, and then optimize the EIG objective that uses this posterior in place of the prior (Ryan et al., 2016)

$$I(\xi_t) = \mathbb{E}_{p(\theta|\xi_{1:t-1}, y_{1:t-1})p(y_t|\theta, \xi_t)} \left[\log \frac{p(y_t|\theta, \xi_t)}{p(y_t|\xi_t)} \right] \quad (3)$$

where $p(y_t|\xi_t) = \mathbb{E}_{p(\theta|\xi_{1:t-1}, y_{1:t-1})} [p(y_t|\theta, \xi_t)]$.

Despite the great potential of the adaptive BOED framework, this conventional approach is very computationally expensive. At each stage t of the experiment we must compute the posterior $p(\theta|\xi_{1:t-1}, y_{1:t-1})$, which is costly and cannot be done in advance as it depends on $y_{1:t-1}$. Furthermore, the posterior is then used to obtain ξ_t by maximizing the objective in (3), which is computationally even more demanding as it involves the optimization of a doubly intractable quantity (Rainforth et al., 2018; Foster et al., 2019). Both of these steps must be done during the experiment, meaning it is infeasible to run adaptive BOED in real time experiment settings unless the model is unusually simple.

2.2. Contrastive information bounds

In Foster et al. (2020), the authors noted that if $\xi \in \Xi$ is continuous, approximate optimization of the EIG at each stage of the experiment can be achieved in a single *unified* stochastic gradient procedure that both estimates and optimizes the EIG simultaneously. A key component of this approach is the derivation of several contrastive lower bounds on the EIG, inspired by work in representation learning (van den Oord et al., 2018; Poole et al., 2019). One such bound is the Prior Contrastive Estimation (PCE) bound, given by

$$I(\xi) \geq \mathbb{E} \left[\log \frac{p(y|\theta_0, \xi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(y|\theta_\ell, \xi)} \right] \quad (4)$$

where $\theta_0 \sim p(\theta)$ is the sample used to generate $y \sim p(y|\theta, \xi)$ and $\theta_{1:L}$ are L contrastive samples drawn independently from $p(\theta)$; as $L \rightarrow \infty$ the bound becomes tight. The PCE bound can be maximized by stochastic gradient ascent (SGA) (Robbins & Monro, 1951) to approximate the optimal design ξ . As discussed previously, in a sequential setting this stochastic gradient optimization is repeated T times, with $p(\theta)$ replaced by $p(\theta|\xi_{1:t-1}, y_{1:t-1})$ at step t .

3. Rethinking Sequential BOED

To enable adaptive BOED to be deployed in settings where design decisions must be taken quickly, we first need to rethink the traditional iterative approach to produce a formulation which considers the entire design process holistically. To this end, we introduce the concept of a *design function*, or *policy*, π that maps from the set of all previous design-observation pairs to the next chosen design.

Let h_t denote the experimental *history* $(\xi_1, y_1), \dots, (\xi_t, y_t)$. We can simulate histories for a given policy π , by sampling

a $\theta \sim p(\theta)$, then, for each $t = 1, \dots, T$, fixing $\xi_t = \pi(h_{t-1})$ (where $h_0 = \emptyset$) and sampling $y_t \sim p(y|\theta, \xi_t)$. The density of this generative process can be written as

$$p(\theta)p(h_T|\theta, \pi) = p(\theta) \prod_{t=1}^T p(y_t|\theta, \xi_t). \quad (5)$$

The standard sequential BOED approach described in § 2.1 now corresponds to a costly implicit policy π_s , that performs posterior estimation followed by EIG optimization to choose each design. By contrast, in DAD, we will learn a deterministic π that chooses designs directly.

Another way to think about π_s is that it is the policy which piecewise optimizes the following objective for $\xi_t|h_{t-1}$

$$I_{h_{t-1}}(\xi_t) := \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t)} \left[\log \frac{p(y_t|\theta, \xi_t)}{p(y_t|h_{t-1}, \xi_t)} \right] \quad (6)$$

where $p(y_t|h_{t-1}, \xi_t) = \mathbb{E}_{p(\theta|h_{t-1})} [p(y_t|\theta, \xi_t)]$. It is thus the optimal *myopic* policy—that is a policy which fails to reason about its own future actions—for an objective given by the sum of EIGs from each experiment iteration. Note that this is not the optimal *overall* policy as it fails to account for future decision making: some designs may allow better future design decisions than others (González et al., 2016; Jiang et al., 2020).¹

Trying to learn an efficient policy that directly mimics π_s would be very computationally challenging because of the difficulties of dealing with both inference and EIG estimation at each iteration of the training. Indeed, the natural way to do this involves running a full, very expensive, simulated sequential BOED process to generate each training example.

We instead propose a novel strategy that reformulates the sequential decision problem in a way that completely eliminates the need for calculating either posterior distributions or intermediate EIGs, while also allowing for non-myopic policies to be learned. This is done by exploiting an important property of the EIG: the total EIG of a sequential experiment is the sum of the (conditional) EIGs for each experiment iteration. This is formalized in the following result, which provides a single expression for the expected information gained from the entire sequence of T experiments.

Theorem 1. *The total expected information gain for policy π over a sequence of T experiments is*

$$\mathcal{I}_T(\pi) := \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \right] \quad (7)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi) - \log p(h_T|\pi)] \quad (8)$$

where $p(h_T|\pi) = \mathbb{E}_{p(\theta)} [p(h_T|\theta, \pi)]$.

¹To give an intuitive example, consider the problem of placing two breakpoints on the line $[0, 1]$ to produce the most evenly sized segments. The optimal myopic policy places its first design at $1/2$ and its second at either $1/4$ or $3/4$. This is suboptimal since the best strategy is to place the two breakpoints at $1/3$ and $2/3$.

The proof is given in Appendix A. Intuitively, $\mathcal{I}_T(\pi)$ is the expected reduction in entropy from the prior $p(\theta)$ to the final posterior $p(\theta|h_T)$, without considering the intermediate posteriors at all. Note here a critical change from previous BOED formulations: $\mathcal{I}_T(\pi)$ is a function of the policy, not the designs themselves, with the latter now being random variables (due to their dependence on previous outcomes) that we take an expectation over. This is actually a strict generalization of conventional BOED frameworks: static design corresponds to policy that consists of T fixed designs with no adaptivity, for which (8) coincides with $I(\xi_{1:T})$, while conventional adaptive BOED approximates π_s .

By reformulating our objective in terms of a policy, we have constructed a single end-to-end objective for adaptive, non-myopic design and which requires negligible computation at deployment time: once π is learned, it can just be directly evaluated during the experiment itself.

4. Deep Adaptive Design

Theorem 1 showed that the optimal design function $\pi^* = \arg \max_{\pi} \mathcal{I}_T(\pi)$ is the one which maximizes the mutual information between the unknown latent θ and the full rollout of histories produced using that policy, h_T . DAD looks to approximate π^* explicitly using a neural network, which we now refer to as the *design network* π_{ϕ} , with trainable parameters ϕ . This policy-based approach marks a major break from existing methods, which do not represent design decisions explicitly as a function, but instead optimize designs on the fly during the experiment.

DAD amortizes the cost of experimental design—by training the network parameters ϕ , the design network is taught to make correct design decisions across a wide range of possible experimental outcomes. This removes the cost of adaptation for the live experiment itself: during deployment the design network will select the next design nearly instantaneously with a single forward pass of the network. Further, it offers a simplification and streamlining of the sequential BOED process: it only requires the upfront end-to-end training of a single neural network and thus negates the need to set up complex *automated* inference and optimization schemes that would otherwise have to run in the background during a live experiment. A high-level summary of the DAD approach is given in Algorithm 1.

Two key technical challenges still stand in the way of realizing the potential of adaptive BOED in real time. First, whilst the unified objective $\mathcal{I}_T(\pi)$ does not require the computation of intermediate posterior distributions, it remains an intractable objective due to the presence of $p(h_T|\pi)$. To deal with this, we derive a family of lower bounds that are appropriate for the policy-based setting and use them to construct stochastic gradient training schemes for ϕ . Second, to ensure that this network can efficiently learn a mapping

Algorithm 1 Deep Adaptive Design (DAD)

Input: Prior $p(\theta)$, likelihood $p(y|\theta, \xi)$, number of steps T

Output: Design network π_{ϕ}

while training compute budget not exceeded **do**

 Sample $\theta_0 \sim p(\theta)$ and set $h_0 = \emptyset$

for $t = 1, \dots, T$ **do**

 Compute $\xi_t = \pi_{\phi}(h_{t-1})$

 Sample $y_t \sim p(y|\theta_0, \xi_t)$

 Set $h_t = \{(\xi_1, y_1), \dots, (\xi_t, y_t)\}$

end

 Compute estimate for $d\mathcal{L}_T/d\phi$ as per § 4.2

 Update ϕ using stochastic gradient ascent scheme

end

At deployment, π_{ϕ} is fixed, we take $\xi_t = \pi_{\phi}(h_{t-1})$, and each y_t is obtained by running an experiment with ξ_t .

from histories to designs, we require an effective architecture. As we show later, the optimal policy is invariant to the order of the history, and we use this key symmetry to architect an effective design network.

4.1. Contrastive bounds for sequential experiments

Our high-level aim is to train π_{ϕ} to maximize the mutual information $\mathcal{I}_T(\pi_{\phi})$. In contrast to most machine learning tasks, this objective is *doubly* intractable and cannot be directly evaluated or even estimated with a conventional Monte Carlo estimator, except in very special cases (Rainforth et al., 2018). In fact, it is extremely challenging and costly to derive *any unbiased* estimate for it or its gradients. To train π_{ϕ} with stochastic gradient methods, we will therefore introduce and optimize *lower bounds* on $\mathcal{I}_T(\pi_{\phi})$, building on the ideas of § 2.2.

Equation (8) shows that the objective function is the expected logarithm of a ratio of two terms. The first is the likelihood of the history, $p(h_T|\theta, \pi)$, and can be directly evaluated using (5). The second term is an intractable marginal $p(h_T|\pi)$ that is different for each sample of the outer expectation and must thus be estimated separately each time.

Given a sample $\theta_0, h_T \sim p(\theta, h_T|\pi)$, we can perform this estimation by introducing L independent *contrastive* samples $\theta_{1:L} \sim p(\theta)$. We can then approximate the log-ratio in two different ways, depending on whether or not we include θ_0 in our estimate for $p(h_T|\pi)$:

$$g_L(\theta_{0:L}, h_T) = \log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_{\ell}, \pi)} \quad (9)$$

$$f_L(\theta_{0:L}, h_T) = \log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_{\ell}, \pi)}. \quad (10)$$

These functions can both be evaluated by recomputing the likelihood of the history under each of the contrastive samples $\theta_{1:L}$. We note that g cannot exceed $\log(L+1)$, whereas f is potentially unbounded (see Appendix A for a proof).

We now show that using g to approximate the integrand leads to a *lower* bound on the overall objective $\mathcal{I}_T(\pi)$, whilst using f leads to an *upper* bound. During training, we focus on the lower bound, because it does not lead to unbounded ratio estimates and is therefore more numerically stable. We refer to this new lower bound as *sequential PCE* (sPCE).

Theorem 2 (Sequential PCE). *For a design function π and a number of contrastive samples $L \geq 0$, let*

$$\mathcal{L}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T | \pi) p(\theta_{1:L})} [g_L(\theta_{0:L}, h_T)] \quad (11)$$

where $g_L(\theta_{0:L}, h_T)$ is as per (9), and $\theta_0, h_T \sim p(\theta, h_T | \pi)$, and $\theta_{1:L} \sim p(\theta)$ independently. Given minor technical assumptions discussed in the proof, we have²

$$\mathcal{L}_T(\pi, L) \uparrow \mathcal{I}_T(\pi) \text{ as } L \rightarrow \infty \quad (12)$$

at a rate $\mathcal{O}(L^{-1})$.

The proof is presented in Appendix A. For evaluation purposes, it is helpful to pair sPCE with an upper bound, which we obtain by using f as our estimate of the integrand

$$\mathcal{U}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T | \pi) p(\theta_{1:L})} [f_L(\theta_{0:L}, h_T)]. \quad (13)$$

We refer to this bound as sequential Nested Monte Carlo (sNMC). Theorem 4 in Appendix A shows that $\mathcal{U}_T(\pi, L)$ satisfies complementary properties to $\mathcal{L}_T(\pi, L)$. In particular, $\mathcal{L}_T(\pi, L) \leq \mathcal{I}_T(\pi) \leq \mathcal{U}_T(\pi, L)$ and both bounds become monotonically tighter as L increases, becoming exact as $L \rightarrow \infty$ at a rate $\mathcal{O}(1/L)$. We can thus directly control the trade-off between bias in our objective and the computational cost of training. Note that increasing L has no impact on the cost at deployment time. Critically, as we will see in our experiments, we tend to only need relatively modest values of L for $\mathcal{L}_T(\pi, L)$ to be an effective objective.

If using a sufficiently large L proves problematic (e.g. our available training time is strictly limited), one can further tighten these bounds for a fixed L by introducing an amortized proposal, $q(\theta; h_T)$, for the contrastive samples $\theta_{1:L}$, rather than drawing them from the prior, as in Foster et al. (2020). By appropriately adapting $\mathcal{L}_T(\pi, L)$, the proposal and the design network can then be trained simultaneously with a single unified objective, in a manner similar to a variational autoencoder (Kingma & Welling, 2014), allowing the bound itself to get tighter during training. The resulting more general class of bounds are described in detail in Appendix B and may offer further improvements for the DAD approach. We focus on training with sPCE here in the interest of simplicity of both exposition and implementation.

4.2. Gradient estimation

The design network parameters ϕ can be optimized using a stochastic optimization scheme such as Adam (Kingma &

² $x_L \uparrow x$ means that x_L is a monotonically increasing sequence in L with limit x .

Ba, 2014). Such methods require us to compute unbiased gradient estimates of the sPCE objective (11). Throughout, we assume that the design space Ξ is continuous.

We first consider the case when the observation space $\mathcal{Y}$ is also continuous and the likelihood $p(y|\theta, \xi)$ is reparametrizable. This means that we can introduce random variables $\epsilon_{1:T} \sim p(\epsilon)$, which are independent of $\xi_{1:T}$ and $\theta_{0:L}$, such that $y_t = y(\theta_0, \xi_t, \epsilon_t)$. As we already have that $\xi_t = \pi_\phi(h_{t-1})$, we see that h_t becomes a deterministic function of h_{t-1} given ϵ_t and θ_0 . Under these assumptions we can take the gradient operator inside the expectation and apply the law of the unconscious statistician to write³

$$\frac{d\mathcal{L}_T}{d\phi} = \mathbb{E}_{p(\theta_{0:L})p(\epsilon_{1:T})} \left[\frac{d}{d\phi} g_L(\theta_{0:L}, h_T) \right]. \quad (14)$$

We can now construct unbiased gradient estimates by sampling from $p(\theta_{0:L})p(\epsilon_{1:T})$ and evaluating, $dg_L(\theta_{0:L}, h_T)/d\phi$. This gradient can be easily computed via an automatic differentiation framework (Baydin et al., 2018; Paszke et al., 2019).

For the case of discrete observations $y \in \mathcal{Y}$, first note that given a policy π_ϕ , the only randomness in the history h_T comes from the observations $y_1, \dots, y_T$, since the designs are computed deterministically from past histories. One approach to computing the gradient of (11) in this case is to sum over all possible histories h_T , integrating out the variables $y_{1:T}$, and take gradients with respect to ϕ to give

$$\frac{d\mathcal{L}_T}{d\phi} = \mathbb{E} \left[\sum_{h_T} \frac{d}{d\phi} \left(p(h_T | \theta_0) g_L(\theta_{0:L}, h_T) \right) \right], \quad (15)$$

where the expectation is over $\theta_{0:L} \sim p(\theta)$. Unbiased gradient estimates can be computed using samples from the prior. Unfortunately, this gradient estimator has a computational cost $\mathcal{O}(|\mathcal{Y}|^T)$ and is therefore only applicable when both the number of experiments T and the number of possible outcomes $|\mathcal{Y}|$ are relatively small.

To deal with the cases when it is either impractical to enumerate all possible histories, or $\mathcal{Y}$ is continuous but the likelihood $p(h_T|\theta, \pi_\phi)$ is non-reparametrizable, we propose using the score function gradient estimator, which is also known as the REINFORCE estimator (Williams, 1992). The score function gradient, is given by

$$\begin{aligned} \frac{d\mathcal{L}_T}{d\phi} = \mathbb{E} \left[\left(\log \frac{p(h_T | \theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T | \theta_\ell, \pi_\phi)} \right) \frac{d}{d\phi} \log p(h_T | \theta_0, \pi_\phi) \right. \\ \left. - \frac{d}{d\phi} \log \sum_{\ell=0}^L p(h_T | \theta_\ell, \pi_\phi) \right] \end{aligned} \quad (16)$$

³We use $\partial a / \partial b$ and da/db to represent the Jacobian matrices of partial and total derivatives respectively for vectors a and b .

where the expectation is over $\theta_0, h_T \sim p(\theta, h_T | \pi)$ and $\theta_{1:L} \sim p(\theta)$, and unbiased estimates may again be obtained using samples. This gradient is amenable to the wide range of existing variance reduction methods such as control variates (Tucker et al., 2017; Mohamed et al., 2020). In our experiments, however, we found the standard score function gradient to be sufficiently low variance. For complete derivations of the gradients estimators we use, see Appendix C.

4.3. Architecture

Finally, we discuss the deep learning architecture used for π_ϕ . To allow efficient and effective training, we take into account a key permutation invariance of the BOED problem as highlighted by the following result (proved in Appendix A).

Theorem 3 (Permutation invariance). *Consider a permutation $\sigma \in S_k$ acting on a history h_k^1 , yielding $h_k^2 = (\xi_{\sigma(1)}, y_{\sigma(1)}), \dots, (\xi_{\sigma(k)}, y_{\sigma(k)})$. For all such σ , we have*

$$\mathbb{E} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \middle| h_k = h_k^1 \right] = \mathbb{E} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \middle| h_k = h_k^2 \right]$$

such that the EIG is unchanged under permutation. Further, the optimal policies starting in h_k^1 and h_k^2 are the same.

This permutation invariance is an important and well-studied property of many machine learning problems (Bloem-Reddy & Teh, 2019). The knowledge that a system exhibits permutation invariance can be exploited in neural architecture design to enable significant *weight sharing*. One common approach is pooling (Edwards & Storkey, 2016; Zaheer et al., 2017; Garnelo et al., 2018a;b). This involves summing or otherwise combining representations of multiple inputs into a single representation that is invariant to their order.

Using this idea, we represent the history h_t with a fixed dimensional representation that is formed by pooling representations of the distinct design-outcome pairs of the history

$$R(h_t) := \sum_{k=1}^t E_{\phi_1}(\xi_k, y_k), \quad (17)$$

where E_{ϕ_1} is a neural network *encoder* with parameters ϕ_1 to be learned. Note that this pooled representation is the same if we reorder the labels $1, \dots, t$. By convention, the sum of an empty sequence is 0.

We then construct our design network to make decisions based on the pooled representation $R(h_t)$ by setting $\pi_\phi(h_t) = F_{\phi_2}(R(h_t))$, where F_{ϕ_2} is a learned *emitter* network. The trainable parameters are $\phi = \{\phi_1, \phi_2\}$. By combining simple networks in a way that is sensitive to the permutation invariance of the problem, we facilitate parameter sharing in which the network E_{ϕ_1} is re-used for each input pair and for each time step t . This results in significantly improved performance compared to networks that are forced to *learn* the relevant symmetries of the problem.

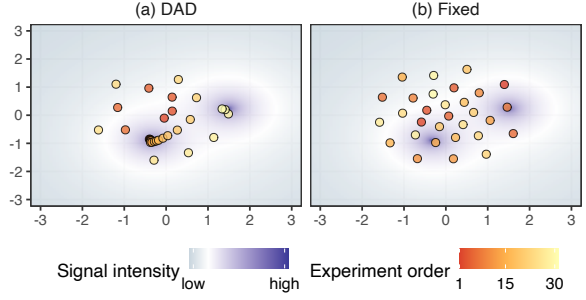


Figure 1. An example of the designs learnt by (a) the DAD network and (b) the fixed baseline for a given θ sampled from the prior.

5. Related Work

Existing approaches to sequential BOED typically follow the path outlined in § 2.1. The posterior inference performed at each stage of the conventional approach has been done using sequential Monte Carlo (Del Moral et al., 2006; Drovandi et al., 2014), population Monte Carlo (Rainforth, 2017), variational inference (Foster et al., 2019; 2020), and Laplace approximation (Lewi et al., 2009; Long et al., 2013).

The estimation of the mutual information objective at each step has been performed by nested Monte Carlo (Myung et al., 2013; Vincent & Rainforth, 2017), variational bounds (Foster et al., 2019; 2020), Laplace approximation (Lewi et al., 2009), ratio estimation (Kleinegesse et al., 2020), and hybrid methods (Senarathne et al., 2020). The optimization over designs has been performed by Bayesian optimization (Foster et al., 2019; Kleinegesse et al., 2020), interacting particle systems (Amzal et al., 2006), simulated annealing (Müller, 2005), utilizing regret bounds (Zheng et al., 2020), or bandit methods (Rainforth, 2017).

There are approaches that simultaneously estimate the mutual information and optimize it, using a single stochastic gradient procedure. Examples include perturbation analysis (Huan & Marzouk, 2014), variational lower bounds (Foster et al., 2020), or multi-level Monte Carlo (Goda et al., 2020).

Some recent work has focused specifically on models with intractable likelihoods (Hainy et al., 2016; Kleinegesse & Gutmann, 2020; Kleinegesse et al., 2020). Other work has sought to learn a non-myopic strategy focusing on specific tractable cases (Huan & Marzouk, 2016; Jiang et al., 2020).

6. Experiments

We now compare **DAD** to a number of baselines across a range of experimental design problems. We implement DAD by extending PyTorch (Paszke et al., 2019) and Pyro (Bingham et al., 2018) to provide an implementation that is abstracted from the specific problem. Code is publicly available at <https://github.com/ae-foster/dad>.

Method	Lower bound, $\mathcal{L}_{30}$	Upper bound, $\mathcal{U}_{30}$
Random	8.303 ± 0.043	8.322 ± 0.045
Fixed	8.838 ± 0.039	8.914 ± 0.038
DAD	10.926 ± 0.036	12.382 ± 0.095
Variational	8.776 ± 0.143	9.064 ± 0.187

Table 1. Upper and lower bounds on the total EIG, $\mathcal{I}_{30}(\pi)$, for the location finding experiment. Errors indicate ± 1 s.e. estimated over 256 (variational) or 2048 (others) rollouts.

As our aim is to adapt designs in *real-time*, we primarily compare to strategies that are fast at deployment time. The simplest baseline is **random** design, which selects designs uniformly at random. The **fixed** baseline completely ignores the opportunity for adaptation and uses static design to learn a fixed $\xi_1, \dots, \xi_T$ before the experiment. We use the SG-BOED approach of Foster et al. (2020) with the PCE bound to optimize the fixed design $\xi_{1:T}$. We also compare to tailor-made heuristics for particular models as appropriate.

Similarly to the notion of the amortization gap in amortized inference (Cremer et al., 2018), one might initially expect to a drop in performance of DAD compared to conventional (non-amortized) BOED methods that use the traditional iterative approach of § 2.1. To assess this we also consider using the SG-BOED approach of Foster et al. (2020) in a traditional iterative manner to approximate π_s , referring to this as the **variational** baseline, noting this requires significant run-time computation. We also look at several iterative BOED baselines that are specifically tailored to the examples that we choose (Vincent & Rainforth, 2017; Kleingegesse et al., 2020). Perhaps surprisingly, we find that DAD is not only competitive compared to these non-amortized methods, but often outperforms them. This is discussed in § 7.

The first performance metric that we focus on is total EIG, $\mathcal{I}_T(\pi)$. When no direct estimate of $\mathcal{I}_T(\pi)$ is available, we estimate both the sPCE lower bound and sNMC upper bound. We also present the standard error to indicate how the performance varies between different experiment realizations (rollouts). We further consider the deployment time (i.e. the time to run the experiment itself, after pre-training); a critical metric for our aims. Full experiment details are given in Appendix D.

6.1. Location finding in 2D

Inspired by the acoustic energy attenuation model of Sheng & Hu (2005), we consider the problem of finding the locations of multiple hidden sources which each emits a signal whose intensity attenuates according to the inverse-square law. The *total intensity* is a superposition of these signals. The design problem is to choose where to make observations of the total signal to learn the locations of the sources.

We train a DAD network to perform $T = 30$ experiments with $K = 2$ sources. The designs learned by DAD are

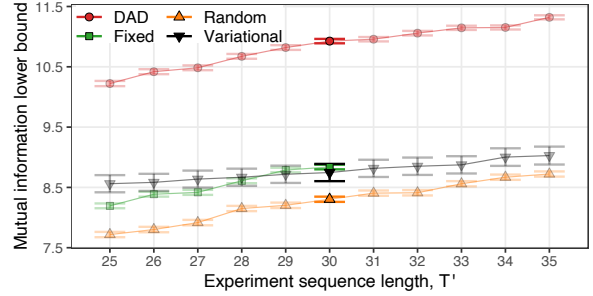


Figure 2. Generalizing sequence length for the location finding experiment. The DAD network and the fixed strategy were trained to perform $T = 30$ experiments, whilst other strategies do not require pre-training. The fixed strategy cannot be generalized to sequences longer than its training regime. We present sPCE estimates with error bars computed as in Table 1.

visualized in Figure 1(a). Here our network learns a complex strategy that initially explores in a spiral pattern. Once it detects a strong signal, multiple experiments are performed close together to refine knowledge of that location (note the high density of evaluations near the sources). The fixed design strategy, displayed in Figure 1(b) must choose all design locations up front, leading to an evenly dispersed strategy that cannot “hone in” on the critical areas, thus gathering less information.

Table 1 reports upper and lower bounds on $\mathcal{I}_T(\pi)$ for each strategy and confirms that DAD significantly outperforms all the considered baselines. DAD is also orders of magnitude faster to deploy than the variational baseline, the other adaptive method, with DAD taking 0.0474 ± 0.0003 secs to make all 30 design decisions on a lightweight CPU, compared to 8963 secs for the variational method.

Varying the Design Horizon In practical situations the exact number of experiments to perform may be unknown. Figure 2 indicates that our DAD network that is pretrained to perform $T = 30$ experiments can generalize well to perform $T' \neq 30$ experiments at deployment time, still outperforming the baselines, indicating that DAD is robust to the length of training sequences.

Training Stability To assess the stability between different training runs, we trained 16 different DAD networks. Computing the mean and standard error of the lower bound on $\mathcal{I}_T(\pi)$ over these 16 runs gave 10.91 ± 0.014 , and the matching upper bounds were 12.47 ± 0.046 . We see that the variance across different training seeds is modest, indicating that DAD reaches designs of a similar quality each time. Comparing with Table 1, we see that the natural variability across rollouts (i.e. different θ) with a single DAD network tends to be larger than the variance between the average performance of different DAD networks.

Method	Deployment time (s)
Frye et al. (2016)	0.0902 ± 0.0003
Kirby (2009)	N/A
Fixed	N/A
DAD	0.0901 ± 0.0007
Badapted	25.2679 ± 0.1854

Table 2. Deployment times for Hyperbolic Temporal Discounting methods. We present the total design time for $T = 20$ questions, taking the mean and ± 1 s.e. over 10 realizations. Tests were conducted on a lightweight CPU (see Appendix D).

Method	Lower bound	Upper bound
Frye et al. (2016)	3.500 ± 0.029	3.513 ± 0.029
Kirby (2009)	1.861 ± 0.008	1.864 ± 0.009
Fixed	2.518 ± 0.007	2.524 ± 0.007
DAD	5.021 ± 0.013	5.123 ± 0.015
Badapted	4.454 ± 0.016	4.536 ± 0.018

Table 3. Final lower and upper bounds on the total information $\mathcal{I}_T(\pi)$ for the Hyperbolic Temporal Discounting experiment. The bounds are finite sample estimates of $\mathcal{L}_T(\pi, L)$ and $\mathcal{U}_T(\pi, L)$ with $L = 5000$. The errors indicate ± 1 s.e. over the sampled histories.

6.2. Hyperbolic temporal discounting

In psychology, temporal discounting is the phenomenon that the utility people attribute to a reward typically decreases the longer they have to wait to receive it (Critchfield & Kollins, 2001; Green & Myerson, 2004). For example, a participant might be willing to trade £90 today for £100 in a month’s time, but not for £100 in a year. A common parametric model for temporal discounting in humans is the hyperbolic model (Mazur, 1987); we study a specific form of this model proposed by Vincent (2016).

We design a sequence of $T = 20$ experiments, each taking the form of a binary question “Would you prefer £ R today, or £100 in D days?” with design $\xi = (R, D)$ that must be chosen at each stage. As real applications of this model would involve human participants, the available time to choose designs is strictly limited. We consider DAD, the aforementioned fixed design policy, and strategies that have been used specifically for experiments of this kind: Kirby (2009), a human constructed fixed set of designs; Frye et al. (2016), a problem-specific adaptive strategy; and Vincent & Rainforth (2017), a partially customized sequential BOED method, called Badapted, that uses population Monte Carlo (Cappé et al., 2004) to approximate the posterior distribution at each step and a bandit approach to optimize the EIG over possible designs.

We begin by investigating the time required to deploy each of these methods. As shown in Table 2, the non-amortized Badapted method takes the longest time, while for DAD, the total deployment time is less than 0.1 seconds—totally imperceptible to a participant.

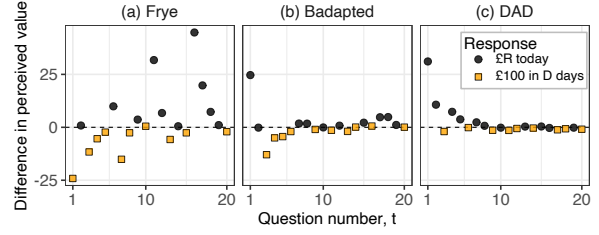


Figure 3. An example of the designs learnt by two of the problem-specific baselines and DAD. We plot the difference in perceived value of the two propositions “£ R today” and “£100 in D days” for a certain participant, represented by a specific value of the latent variable θ . A difference of 0 indicates that the participant is indifferent between the two offers.

Table 3 shows the performance of each method. We see that DAD performs best, surpassing bespoke design methods that have been proposed for this problem, including Badapted which has a considerably larger computation budget. Figure 3 demonstrates how the designs learnt by DAD compare qualitatively with the two most competitive problem-specific baselines. As with Badapted, DAD designs rapidly cluster near the indifference point.

This experiment demonstrates that DAD can successfully amortize the process of experimental design in a real application setting. It outperforms some of the most successful non-amortized and highly problem-specific approaches with a fraction of the cost during the real experiment.

6.3. Death process

We conclude with an example from epidemiology (Cook et al., 2008) in which healthy individuals become infected at rate θ . The design problem is to choose observations times $\xi > 0$ at which to observe the number of infected individuals: we select $T = 4$ designs sequentially with an independent stochastic process observed at each iteration. We compare to our fixed and variational baselines, along with the adaptive SeqBED approach of Kleingesse et al. (2020).

First, we examine the compute time required to deploy each method for a single run of the sequential experiment. The times illustrated in Table 4 show that the adaptive strategy learned by DAD can be deployed in under 0.01 seconds, many orders of magnitude faster than the non-amortized methods, with SeqBED taking hours for *one* rollout.

Next, we estimate the objective $\mathcal{I}_T(\pi)$ by averaging the information gain over simulated rollouts. The results in Table 4 reveal that DAD designs are superior to both fixed design and variational adaptive design, tending to uncover more information about the latent θ across many possible experimental trajectories. For comparison with SeqBED, we were unable to perform sufficient rollouts to obtain a high

Method	Deployment time (s)	$\mathcal{I}_T(\pi)$
Fixed	N/A	2.023 ± 0.007
DAD	$0.0051 \pm 12\%$	2.113 ± 0.008
Variational	$1935.0 \pm 2\%$	2.076 ± 0.034
SeqBED*	25911.0	1.590

Table 4. Total EIG $\mathcal{I}_T(\pi)$ and deployment times for the Death Process. We present the EIG ± 1 s.e. over 10,000 rollouts (fixed and DAD), 500 rollouts (variational) or *1 rollout (SeqBED). The IG can be efficiently evaluated in this case (see Appendix D). Runtimes computed as per Table 2.

quality estimate of $\mathcal{I}_T(\pi)$. Instead, we conducted a single rollout of each method with $\theta = 1.5$ fixed. The resulting information gains for this one rollout were: 1.590 (SeqBED), 1.719 (Variational), 1.678 (Fixed), **1.779 (DAD)**.

7. Discussion

In this paper we introduced DAD—a new method utilizing the power of deep learning to amortize the cost of sequential BOED and allow adaptive experiments to be run in real time. In all experiments DAD performed significantly better than baselines with a comparable deployment time. Further, DAD showed competitive performance against conventional BOED approaches that do not use amortization, but make costly computations at each stage of the experiment.

Surprisingly, we found DAD was often able to outperform these non-amortized approaches despite using a tiny fraction of the resources at deployment time. We suggest two reasons for this. Firstly, conventional methods must approximate the posterior $p(\theta|h_t)$ at each stage. If this approximation is poor, the resulting design optimization will yield poor results regardless of the EIG optimization approach chosen. Careful tuning of the posterior approximation could alleviate this, but would increase computational time further and it is difficult to do this in the required automated manner. DAD sidesteps this problem altogether by eliminating the need for directly approximating a posterior distribution.

Secondly, the policy learnt by DAD has the potential to be *non-myopic*: it does not choose a design that is optimal for the current experiment in isolation, but takes into account the fact that there are more experiments to be performed in the future. We can see this in practice in a simple experiment using the location finding example with one source in 1D with prior $\theta \sim N(0, 1)$ and with $T = 2$ steps. This setting is simple enough to compute the *exact* one-step optimal design via numerical integration. Figure 4 [Left] shows the design function learnt by DAD alongside the exact optimal myopic design. The optimal myopic strategy for $t = 1$ is to sample at the prior mean $\xi_1 = 0$. At time $t = 2$ the myopic strategy selects a positive or negative design with equal probability. In contrast, the policy learnt by DAD is to sample at $\xi_1 \approx -0.4$, which does not optimize the EIG

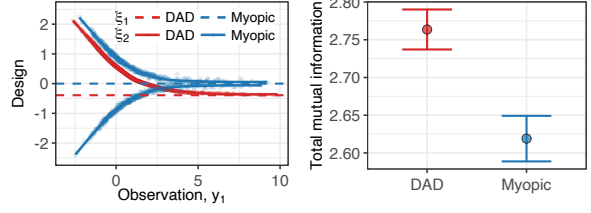


Figure 4. 1D location finding with 1 source, $T = 2$. [Left] the design function, dashed lines correspond to the first design ξ_1 , which is independent of y_1 . [Right] $\mathcal{I}_2(\pi)$, the total EIG ± 1 s.e.

for $T = 1$ in isolation, but leads to a better *overall* design strategy that focuses on searching the positive regime $\xi_2 > \xi_1$ in the second experiment. Figure 4 [Right] confirms that the policy learned by DAD achieves higher total EIG from the two step experiment than the *exact* myopic approach.

Limitations and Future Work The present form of DAD still possesses some restrictions that future work might look to address. Firstly, it requires the likelihood model to be *explicit*, i.e. that we can evaluate the density $p(y_t|\theta, \xi_t)$. Secondly, it requires the experiments to be conditionally independent given θ , i.e. $p(y_{1:T}|\theta, \xi_{1:T}) = \prod_{t=1}^T p(y_t|\theta, \xi_t)$, which may not be the case for, e.g. time series models. Thirdly, it requires the designs themselves, ξ_t , to be continuous to allow for gradient-based optimization. On another note, DAD’s use of a policy to make design decisions establishes a critical link between experimental design and model-based reinforcement learning (Sekar et al., 2020). Though DAD is distinct in several important ways (e.g. the lack of observed rewards), investigating these links further might provide an interesting avenue for future work.

Conclusions DAD represents a new conception of adaptive experimentation that focuses on learning a design *policy* network offline, then deploying it during the live experiment to quickly make adaptive design decisions. This marks a departure from the well-worn path of myopic adaptive BOED (Sec. 2), eliminating the need to estimate intermediate posterior distributions or optimize over designs during the live experiment itself; it represents the first approach to allow adaptive BOED to be run in real-time for general problems. As such, we believe it may be beneficial to practitioners in a number of fields, from online surveys to clinical trials.

Acknowledgements

AF gratefully acknowledges funding from EPSRC grant no. EP/N509711/1. DRI is supported by EPSRC through the Modern Statistics and Statistical Machine Learning (StatML) CDT programme, grant no. EP/S023151/1. AF would like to thank Martin Jankowiak and Adam Golinski for helpful discussions about amortizing BOED.

References

- Amzal, B., Bois, F. Y., Parent, E., and Robert, C. P. Bayesian-optimal design via interacting particle systems. *Journal of the American Statistical association*, 101(474):773–785, 2006.
- Angelova, J. A. On moments of sample mean and variance. *Int. J. Pure Appl. Math*, 79(1):67–85, 2012.
- Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18, 2018.
- Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 2018.
- Bloem-Reddy, B. and Teh, Y. W. Probabilistic symmetry and invariant neural networks. *arXiv preprint arXiv:1901.06082*, 2019.
- Cappé, O., Guillin, A., Marin, J.-M., and Robert, C. P. Population monte carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.
- Chaloner, K. and Verdinelli, I. Bayesian experimental design: A review. *Statistical Science*, pp. 273–304, 1995.
- Cook, A. R., Gibson, G. J., and Gilligan, C. A. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008.
- Cremer, C., Li, X., and Duvenaud, D. Inference suboptimality in variational autoencoders. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1078–1086. PMLR, 2018.
- Critchfield, T. S. and Kollins, S. H. Temporal discounting: Basic research and the analysis of socially important behavior. *Journal of applied behavior analysis*, 34(1): 101–122, 2001.
- Del Moral, P., Doucet, A., and Jasra, A. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.
- Drovandi, C. C., McGree, J. M., and Pettitt, A. N. A sequential monte carlo algorithm to incorporate model uncertainty in bayesian sequential design. *Journal of Computational and Graphical Statistics*, 23(1):3–24, 2014.
- Dushenko, S., Ambal, K., and McMichael, R. D. Sequential bayesian experiment design for optically detected magnetic resonance of nitrogen-vacancy centers. *Physical Review Applied*, 14(5):054036, 2020.
- Edwards, H. and Storkey, A. Towards a neural statistician. *arXiv preprint arXiv:1606.02185*, 2016.
- Evans, J. R. and Mathur, A. The value of online surveys. *Internet research*, 2005.
- Foster, A., Jankowiak, M., Bingham, E., Horsfall, P., Teh, Y. W., Rainforth, T., and Goodman, N. Variational Bayesian Optimal Experimental Design. In *Advances in Neural Information Processing Systems 32*, pp. 14036–14047. Curran Associates, Inc., 2019.
- Foster, A., Jankowiak, M., O’Meara, M., Teh, Y. W., and Rainforth, T. A unified stochastic gradient approach to designing bayesian-optimal experiments. volume 108 of *Proceedings of Machine Learning Research*, pp. 2959–2969, Online, 26–28 Aug 2020. PMLR.
- Frye, C. C., Galizio, A., Friedel, J. E., DeHart, W. B., and Odum, A. L. Measuring delay discounting in humans using an adjusting amount task. *JoVE (Journal of Visualized Experiments)*, (107):e53584, 2016.
- Garnelo, M., Rosenbaum, D., Maddison, C. J., Ramalho, T., Saxton, D., Shanahan, M., Teh, Y. W., Rezende, D. J., and Eslami, S. Conditional neural processes. *arXiv preprint arXiv:1807.01613*, 2018a.
- Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018b.
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.
- Goda, T., Hironaka, T., and Kitade, W. Unbiased mlmc stochastic gradient-based optimization of bayesian experimental designs. *arXiv preprint arXiv:2005.08414*, 2020.
- González, J., Osborne, M., and Lawrence, N. Glasses: Relieving the myopia of bayesian optimisation. In *Artificial Intelligence and Statistics*, pp. 790–799. PMLR, 2016.
- Green, L. and Myerson, J. A discounting framework for choice with delayed and probabilistic rewards. *Psychological bulletin*, 130(5):769, 2004.
- Hainy, M., Drovandi, C. C., and McGree, J. M. Likelihood-free extensions for bayesian sequentially designed experiments. In Kunert, J., Müller, C. H., and Atkinson, A. C. (eds.), *mODA 11 - Advances in Model-Oriented Design and Analysis*, pp. 153–161. Springer International Publishing, 2016.
- Huan, X. and Marzouk, Y. Gradient-based stochastic optimization methods in bayesian experimental design. *International Journal for Uncertainty Quantification*, 4(6), 2014.

- Huan, X. and Marzouk, Y. M. Sequential bayesian optimal experimental design via approximate dynamic programming. *arXiv preprint arXiv:1604.08320*, 2016.
- Jiang, S., Chai, H., Gonzalez, J., and Garnett, R. Binoculars for efficient, nonmyopic sequential experimental design. In *International Conference on Machine Learning*, pp. 4794–4803. PMLR, 2020.
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Kingma, D. P. and Welling, M. Auto-encoding variational Bayes. In *ICLR*, 2014.
- Kirby, K. N. One-year temporal stability of delay-discount rates. *Psychonomic bulletin & review*, 16(3):457–462, 2009.
- Kleinegesse, S. and Gutmann, M. 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, pp. 5316–5326. PMLR, 2020. URL <http://proceedings.mlr.press/v119/kleinegesse20a.html>.
- Kleinegesse, S., Drovandi, C., and Gutmann, M. U. Sequential bayesian experimental design for implicit models via mutual information. *arXiv preprint arXiv:2003.09379*, 2020.
- Kruschke, J. Doing bayesian data analysis: A tutorial with r, jags, and stan. 2014.
- Lewi, J., Butera, R., and Paninski, L. Sequential optimal design of neurophysiology experiments. *Neural Computation*, 21(3):619–687, 2009.
- Lindley, D. V. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pp. 986–1005, 1956.
- Long, Q., Scavino, M., Tempone, R., and Wang, S. Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations. *Computer Methods in Applied Mechanics and Engineering*, 259:24–39, 2013.
- Lyu, J., Wang, S., Balias, T. E., Singh, I., Levit, A., Moroz, Y. S., OMeara, M. J., Che, T., Algaa, E., Tolmachova, K., et al. Ultra-large library docking for discovering new chemotypes. *Nature*, 566(7743):224, 2019.
- Mazur, J. E. An adjusting procedure for studying delayed reinforcement. *Commons, ML.; Mazur, JE.; Nevin, JA*, pp. 55–73, 1987.
- Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.
- Müller, P. Simulation based optimal design. *Handbook of Statistics*, 25:509–518, 2005.
- Myung, J. I., Cavagnaro, D. R., and Pitt, M. A. A tutorial on adaptive design optimization. *Journal of mathematical psychology*, 57(3-4):53–67, 2013.
- Nowozin, S. Debiasing evidence approximations: On importance-weighted autoencoders and jackknife variational inference. In *International Conference on Learning Representations*, 2018.
- Pasek, J. and Krosnick, J. A. Optimizing survey questionnaire design in political science. In *The Oxford handbook of American elections and political behavior*. 2010.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.
- Poole, B., Ozair, S., van den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, pp. 5171–5180, 2019.
- Rainforth, T. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, University of Oxford, 2017.
- Rainforth, T., Cornish, R., Yang, H., Warrington, A., and Wood, F. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pp. 4267–4276. PMLR, 2018.
- Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.
- Ryan, E. G., Drovandi, C. C., McGree, J. M., and Pettitt, A. N. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016.
- Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pp. 8583–8592. PMLR, 2020.

- Senarathne, S., Drovandi, C., and McGree, J. A laplace-based algorithm for bayesian adaptive design. *Statistics and Computing*, 30(5):1183–1208, 2020.
- Sheng, X. and Hu, Y. H. 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.
- Stuhlmüller, A., Taylor, J., and Goodman, N. Learning stochastic inverses. In *Advances in neural information processing systems*, pp. 3048–3056, 2013.
- Tucker, G., Mnih, A., Maddison, C. J., Lawson, D., and Sohl-Dickstein, J. Rebar: Low-variance, unbiased gradient estimates for discrete latent variable models. *arXiv preprint arXiv:1703.07370*, 2017.
- van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Vanlier, J., Tiemann, C. A., Hilbers, P. A., and van Riel, N. A. A Bayesian approach to targeted experiment design. *Bioinformatics*, 28(8):1136–1142, 2012.
- Vincent, B. T. Hierarchical bayesian estimation and hypothesis testing for delay discounting tasks. *Behavior research methods*, 48(4):1608–1620, 2016.
- Vincent, B. T. and Rainforth, T. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. 2017.
- Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992. ISSN 0885-6125. doi: 10.1007/bf00992696.
- Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R., and Smola, A. Deep sets. *arXiv preprint arXiv:1703.06114*, 2017.
- Zheng, S., Pacheco, J., and Fisher, J. A robust approach to sequential information theoretic planning. In *International Conference on Machine Learning*, pp. 5941–5949, 2018.
- Zheng, S., Hayden, D., Pacheco, J., and Fisher III, J. W. Sequential bayesian experimental design with variable cost structure. *Advances in Neural Information Processing Systems*, 33, 2020.

A. Proofs

We begin by showing that $g_L(\theta_{0:L}, h_T)$ from equation (9) is bounded by $\log(L + 1)$ and that $f_L(\theta_{0:L}, h_T)$ from equation (10) can potentially be unbounded. For the former

$$g_L(\theta_{0:L}, h_T) = \log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)} \quad (18)$$

$$= \log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\theta_0, \pi) + \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} + \log(L + 1) \quad (19)$$

$$\leq \log(1) + \log(L + 1). \quad (20)$$

For the latter we have

$$f_L(\theta_{0:L}, h_T) = \log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} \rightarrow +\infty \text{ as } \max_{1 \leq \ell \leq L} p(h_T|\theta_\ell, \pi) \rightarrow 0 \text{ with } p(h_T|\theta_0, \pi) \text{ held constant.}$$

Next we present proofs for all Theorems in the main paper, with each restated for convenience.

Theorem 1. *The total expected information gain for policy π over a sequence of T experiments is*

$$\mathcal{I}_T(\pi) := \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \right] \quad (7)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi) - \log p(h_T|\pi)] \quad (8)$$

where $p(h_T|\pi) = \mathbb{E}_{p(\theta)}[p(h_T|\theta, \pi)]$.

Proof. We begin by rewriting $I_{h_{t-1}}$ in terms of the information gain. This closely mimics the development that we presented in Section 2. By repeated application of Bayes Theorem we have

$$I_{h_{t-1}}(\xi_t) = \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t)} \left[\log \frac{p(y_t|\theta, \xi_t)}{p(y_t|h_{t-1}, \xi_t)} \right] \quad (21)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t)} \left[\log \frac{p(\theta|h_{t-1})p(y_t|\theta, \xi_t)}{p(\theta|h_{t-1})p(y_t|h_{t-1}, \xi_t)} \right] \quad (22)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t)} \left[\log \frac{p(\theta|h_{t-1}, \xi_t, y_t)}{p(\theta|h_{t-1})} \right] \quad (23)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})} [-\log p(\theta|h_{t-1})] + \mathbb{E}_{p(y_t, \theta|\xi_t, h_{t-1})} [\log p(\theta|h_{t-1}, \xi_t, y_t)] \quad (24)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})} [-\log p(\theta|h_{t-1})] + \mathbb{E}_{p(y_t|\xi_t, h_{t-1})p(\theta|h_{t-1}, \xi_t, y_t)} [\log p(\theta|h_{t-1}, \xi_t, y_t)] \quad (25)$$

$$= \mathbb{E}_{p(y_t|\xi_t, h_{t-1})} [H[p(\theta|h_{t-1})] - H[p(\theta|h_{t-1}, \xi_t, y_t)]] \quad (26)$$

Now noting that each $I_{h_{t-1}}(\xi_t)$ is completely determined by h_{t-1} and π (in particular noting that ξ_t is deterministic given these, while θ is already marginalized out in each $I_{h_{t-1}}(\xi_t)$), we can write

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(h_T|\pi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \right] \quad (27)$$

$$= \sum_{t=1}^T \mathbb{E}_{p(h_{t-1}|\pi)} [I_{h_{t-1}}(\xi_t)] \quad (28)$$

and substituting in our earlier formulation for $I_{h_{t-1}}(\xi_t)$

$$= \sum_{t=1}^T \mathbb{E}_{p(h_{t-1}|\pi)} [\mathbb{E}_{p(y_t|\xi_t, h_{t-1})} [H[p(\theta|h_{t-1})] - H[p(\theta|h_{t-1}, \xi_t, y_t)]]] \quad (29)$$

We now observe that we can write $h_t = h_{t-1} \cup \{(\xi_t, y_t)\}$, which allows us to rewrite this as

$$= \sum_{t=1}^T \mathbb{E}_{p(h_t|\pi)} [H[p(\theta|h_{t-1})] - H[p(\theta|h_t)]] \quad (30)$$

$$= \sum_{t=1}^T \mathbb{E}_{p(h_T|\pi)} [H[p(\theta|h_{t-1})] - H[p(\theta|h_t)]] \quad (31)$$

$$= \mathbb{E}_{p(h_T|\pi)} \left[\sum_{t=1}^T H[p(\theta|h_{t-1})] - H[p(\theta|h_t)] \right] \quad (32)$$

$$= \mathbb{E}_{p(h_T|\pi)} [H[p(\theta)] - H[p(\theta|h_T)]], \quad (33)$$

where the last line follows from the fact that we have a telescopic sum. To complete the proof, we rearrange this as

$$= \mathbb{E}_{p(\theta, h_T|\pi)} [\log p(\theta|h_T) - \log p(\theta)] \quad (34)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(\theta)p(h_T|\theta, \pi)}{p(h_T|\pi)} - \log p(\theta) \right] \quad (35)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi) - \log p(h_T|\pi)] \quad (36)$$

as required. $\square$

Theorem 2 (Sequential PCE). *For a design function π and a number of contrastive samples $L \geq 0$, let*

$$\mathcal{L}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T|\pi)p(\theta_{1:L})} [g_L(\theta_{0:L}, h_T)] \quad (11)$$

where $g_L(\theta_{0:L}, h_T)$ is as per (9), and $\theta_0, h_T \sim p(\theta, h_T|\pi)$, and $\theta_{1:L} \sim p(\theta)$ independently. Given minor technical assumptions discussed in the proof, we have⁴

$$\mathcal{L}_T(\pi, L) \uparrow \mathcal{I}_T(\pi) \text{ as } L \rightarrow \infty \quad (12)$$

at a rate $\mathcal{O}(L^{-1})$.

Proof. We first show that $\mathcal{L}_T(\pi, L)$ is a lower bound on $\mathcal{I}_T(\pi)$:

$$\mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] - \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)} \right] \quad (37)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right] \quad (38)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \left(\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(\theta_\ell|h_T)}{p(\theta_\ell)} \right) \right] \quad (39)$$

now introducing the shorthand $p(\theta_{0:L}^{-\ell}) := p(\theta_{0:L \setminus \{\ell\}}) = \prod_{j=0, j \neq \ell}^L p(\theta_j)$,

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell|h_T)p(\theta_{0:L}^{-\ell})}{p(\theta_{0:L})} \right]. \quad (40)$$

⁴ $x_L \uparrow x$ means that x_L is a monotonically increasing sequence in L with limit x .

Now by the systemtry on term in side the log, we see that this expectation would be the same if it were instead taken over $p(\theta_i, h_T|\pi)p(\theta_{0:L}^{-i})$ for any $i \in \{0, \dots, L\}$ (with $i = 0$ giving the original form). Furthermore, the result is unchanged if we take the expectation over the mixture distribution $\frac{1}{L+1} \sum_{i=0}^L p(\theta_i, h_T|\pi)p(\theta_{0:L}^{-i}) = p(h_T|\pi) \frac{1}{L+1} \sum_{i=0}^L p(\theta_i|h_T)p(\theta_{0:L}^{-i})$ and thus we have

$$= \mathbb{E}_{p(h_T|\pi)} \mathbb{E}_{\frac{1}{L+1} \sum_{i=0}^L p(\theta_i|h_T)p(\theta_{0:L}^{-i})} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell|h_T)p(\theta_{0:L}^{-\ell})}{p(\theta_{0:L})} \right] \quad (41)$$

$$= \mathbb{E}_{p(h_T|\pi)} [\text{KL}(\tilde{p}(\theta_{0:L}|h_T)||p(\theta_{0:L}))] \quad (42)$$

where $\tilde{p}(\theta_{0:L}|h_T) = \frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell|h_T)p(\theta_{0:L}^{-\ell})$, which is indeed a distribution since

$$\int \tilde{p}(\theta_{0:L}|h_T) d\theta_{0:L} = \frac{1}{L+1} \sum_{\ell=0}^L \left(\int p(\theta_\ell|h_T) d\theta_\ell \cdot \int p(\theta_{0:L}^{-\ell}) d\theta_{0:L}^{-\ell} \right) = 1. \quad (43)$$

Now by Gibbs' inequality the expected KL in (42) must be non-negative, establishing $\mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) \geq 0$ and thus $\mathcal{I}_T(\pi) \geq \mathcal{L}_T(\pi, L)$ as required.

We next show monotonicity in L , i.e. $\mathcal{L}_T(\pi, L_2) \geq \mathcal{L}_T(\pi, L_1)$ for $L_2 \geq L_1 \geq 0$, using similar argument as above

$$\mathcal{L}_T(\pi, L_2) - \mathcal{L}_T(\pi, L_1) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{\frac{1}{L_1+1} \sum_{i=0}^{L_1} p(h_T|\theta_i, \pi)}{\frac{1}{L_2+1} \sum_{j=0}^{L_2} p(h_T|\theta_j, \pi)} \right] \quad (44)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{\frac{1}{L_1+1} \sum_{i=0}^{L_1} (p(\theta_i|h_T)/p(\theta_i))}{\frac{1}{L_2+1} \sum_{j=0}^{L_2} (p(\theta_j|h_T)/p(\theta_j))} \right] \quad (45)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{\frac{1}{L_1+1} \sum_{i=0}^{L_1} (p(\theta_i|h_T)p(\theta_{0:L_1}^{-i}))/p(\theta_{0:L_1})}{\frac{1}{L_2+1} \sum_{j=0}^{L_2} (p(\theta_j|h_T)p(\theta_{0:L_2}^{-j}))/p(\theta_{0:L_2})} \right] \quad (46)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{\frac{1}{L_1+1} \sum_{i=0}^{L_1} p(\theta_i|h_T)p(\theta_{0:L_1}^{-i})}{\frac{1}{L_2+1} \sum_{j=0}^{L_2} p(\theta_j|h_T)p(\theta_{0:L_2}^{-j})} \right] \quad (47)$$

$$= \mathbb{E}_{p(h_T|\pi)} \mathbb{E}_{\frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell|h_T)p(\theta_{0:L}^{-\ell})} \left[\log \frac{\frac{1}{L_1+1} \sum_{i=0}^{L_1} p(\theta_i|h_T)p(\theta_{0:L_1}^{-i})}{\frac{1}{L_2+1} \sum_{j=0}^{L_2} p(\theta_j|h_T)p(\theta_{0:L_2}^{-j})} \right] \quad (48)$$

$$= \mathbb{E}_{p(h_T|\pi)} [\text{KL}(\tilde{p}_1||\tilde{p}_2)] \geq 0 \quad (49)$$

where $\tilde{p}_1$ and $\tilde{p}_2$ are, respectively, the distributions in the numerator and denominator in (48). The result then again follows by Gibbs' inequality.

Next we show $\mathcal{L}_T(\pi, L) \rightarrow \mathcal{I}_T(\pi)$ as $L \rightarrow \infty$. First, note that the denominator in (11), $\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)$, is a consistent estimator of the marginal $p(h_T|\pi)$, since $\frac{1}{L+1} p(h_T|\theta_0, \pi) \rightarrow 0$, and by the Strong Law of Large Numbers

$$\frac{1}{L+1} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi) = \frac{L}{L+1} \cdot \frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi) \xrightarrow{\text{a.s.}} \mathbb{E}_{p(\theta)} [p(h_T|\theta, \pi)] = p(h_T|\pi). \quad (50)$$

Now from (38) we also have that

$$\mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right] \quad (51)$$

and we have $\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \rightarrow 0$ almost surely as $L \rightarrow \infty$. The minor technical assumption, which is required to

establish convergence is that there exist some $0 < \kappa_1, \kappa_2 < \infty$ such that⁵

$$\kappa_1 \leq \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \leq \kappa_2 \quad \forall \theta, h_T. \quad (52)$$

using this assumption, the integrand of (51) is bounded, because

$$\left| \log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right| = \left| \log \left(\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right) \right| \quad (53)$$

$$\leq \max \left(\left| \log \left(\max_{\ell} \frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right) \right|, \left| \log \left(\min_{\ell} \frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right) \right| \right) \quad (54)$$

$$\leq \max(|\log \kappa_2|, |\log \kappa_1|) \quad (55)$$

$$< \infty. \quad (56)$$

Thus, the Bounded Convergence Theorem can be applied to conclude that $\mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) \rightarrow 0$ as $L \rightarrow \infty$.

Finally, for the rate of convergence we apply the inequality $\log x \leq x - 1$ to (38) to get

$$\mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right] \quad (57)$$

$$\leq \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\frac{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} - 1 \right] \quad (58)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\frac{\frac{1}{L+1} \left(p(h_T|\theta_0\pi) + \sum_{\ell=1}^L \mathbb{E}_{p(\theta_{1:L})} [p(h_T|\theta_\ell, \pi)] \right)}{p(h_T|\pi)} - 1 \right] \quad (59)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\frac{\frac{1}{L+1} (p(h_T|\theta_0\pi) + Lp(h_T|\pi))}{p(h_T|\pi)} - 1 \right] \quad (60)$$

$$= \frac{1}{L+1} \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} - 1 \right] \quad (61)$$

$$= \frac{C}{L+1}, \quad (62)$$

where we can conclude $C < \infty$ using (52). Combining this with the our previous result showing that $\mathcal{L}_T(\pi, L)$ is a lower bound on $\mathcal{I}_T(\pi)$, we have shown that

$$0 \leq \mathcal{I}_T(\pi) - \mathcal{L}_T(\pi, L) \leq \frac{C}{L+1}. \quad (63)$$

This establishes the $\mathcal{O}(L^{-1})$ rate of convergence. $\square$

Theorem 3 (Permutation invariance). *Consider a permutation $\sigma \in S_k$ acting on a history h_k^1 , yielding $h_k^2 = (\xi_{\sigma(1)}, y_{\sigma(1)}), \dots, (\xi_{\sigma(k)}, y_{\sigma(k)})$. For all such σ , we have*

$$\mathbb{E} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \middle| h_k = h_k^1 \right] = \mathbb{E} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \middle| h_k = h_k^2 \right]$$

such that the EIG is unchanged under permutation. Further, the optimal policies starting in h_k^1 and h_k^2 are the same.

Technical note: In this statement, the first expectation is with respect to $p(h_T|\pi)$ for policy π and the second is with respect to $p(h_T|\pi')$, where for $t > k$ we set $\pi'(h_t) = \pi(\sigma^{-1}(h_t))$ where σ^{-1} acts on the first k labels by permutation and as the identity on other labels. This means we remove *explicit* variability under permutation caused by π , and show that no other source of variability can arise.

⁵In practice, we can actually weaken this assumption significantly if necessary by making κ_1 and κ_2 dependent on h_T and θ then assuming that the expectation $\mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} [\log |\kappa_i(\theta_j, h_T)|]$ is finite for $i \in \{1, 2\}$ and $j \in \{0, 1\}$. This then permits $\kappa_1(h_T, \theta) \rightarrow 0$ and $\kappa_2(h_T, \theta) \rightarrow \infty$ for certain h_T and θ , provided that these events are zero measure under both $p(\theta, h_T|\pi)$ and $p(\theta)p(h_T|\pi)$, thereby avoiding potential issues with tail behavior in the limits of extreme values for θ .

Proof. To begin, we set up some notation. Given the partial history $h_k = h_k^1$, we complete the experiment by sampling (ξ_t, y_t) for $t = k + 1, \dots, T$. We denote the resulting full history as h_T^1 , and define h_T^2 similarly. Next, we use Theorem 1 to rewrite the conditional objective under consideration as

$$\mathbb{E}_{p(h_T^1|\pi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \middle| h_k = h_k^1 \right] = \mathbb{E}_{p(\theta|h_k^1) \prod_{t=k+1}^T p(y_t|\theta, \xi_t)} [\log p(h_T^1|\theta, \pi) - \log p(h_T^1|\pi)] \quad (64)$$

$$= \mathbb{E}_{p(\theta|h_k^1) \prod_{t=k+1}^T p(y_t|\theta, \xi_t)} [\log p(\theta|h_T^1) - \log p(\theta)] \quad (65)$$

$$= \mathbb{E}_{p(\theta|h_k^1)p(h_T^1|h_k^1, \theta, \pi)} [\log p(\theta|h_T^1) - \log p(\theta)]. \quad (66)$$

A central point of the proof is that the posterior distribution $p(\theta|h_t)$ is invariant to the order of the history. Indeed, we have

$$p(\theta|h_t) \propto p(\theta) \prod_{s=1}^t p(y_s|\theta, \xi_s) \quad (67)$$

which shows that $p(\theta|h_k^1) = p(\theta|h_k^2)$. Given a continuation of the history $(\xi_{k+1}, y_{k+1}), \dots, (\xi_T, y_T)$, if we use the same continuation starting from h_k^1 and h_k^2 to give h_T^1 and h_T^2 then we have $p(\theta|h_T^1) = p(\theta|h_T^2)$. However, we need to show that the continuations $(\xi_{k+1}, y_{k+1}), \dots, (\xi_T, y_T)$ are equal in distribution.

We now show that the sampling distributions of $(\xi_{k+1}, y_{k+1}), \dots, (\xi_T, y_T)$ are equal starting from h_k^1 and h_k^2 . We have shown that $\theta \sim p(\theta|h_k^1)$ is unchanged in distribution if we instead sample $\theta \sim p(\theta|h_k^2)$. Further, we have

$$\xi_{k+1}^1 = \pi(h_k^1) \quad \xi_{k+1}^2 = \pi'(h_k^2) \quad (68)$$

which, by the construction of π' implies $\xi_{k+1}^1 = \xi_{k+1}^2$. Together, these results imply that the observations y_{k+1}^1 and y_{k+1}^2 are equal in distribution. Proceeding inductively, since h_{k+1}^1 and h_{k+1}^2 are equal in distribution a similar argument shows that h_{k+2}^1 and h_{k+2}^2 have the same distribution. Continuing in this way, we have that h_T^1 and h_T^2 are equal in distribution. Together, these results imply that

$$\mathbb{E}_{p(\theta|h_k^1)p(h_T^1|h_k^1, \theta, \pi)} [\log p(\theta|h_T^1) - \log p(\theta)] = \mathbb{E}_{p(\theta|h_k^2)p(h_T^2|h_k^2, \theta, \pi')} [\log p(\theta|h_T^2) - \log p(\theta)] \quad (69)$$

which conclude the first part of the proof.

To establish the permutation invariance of the optimal policy π^* , we reason by induction starting with $k = T - 1$, using a dynamic programming style argument. Given h_{T-1} , the total EIG is a function of $p(\theta|h_{T-1})$ and ξ_T . Since we do not need to account for future asymmetry in the policy, we immediately have that the optimal final design ξ_T only depends on $p(\theta|h_{T-1})$, which implies that is invariant to the order of the history.

We now assume that the optimal policy is permutation invariant starting from $k + 2$. Using the previous result (69), we separate out the design ξ_{k+1} and substitute π^* for both π and π' (since it is permutation invariant for the steps after $k + 1$ by inductive hypothesis) to give

$$\begin{aligned} & \mathbb{E}_{p(\theta|h_k^1)p(y_{k+1}|\theta, \xi_{k+1}) \prod_{t=k+2}^T p(y_t|\theta, \pi^*(h_{t-1}))} [\log p(\theta|h_T^1) - \log p(\theta)] \\ &= \mathbb{E}_{p(\theta|h_k^2)p(y_{k+1}|\theta, \xi_{k+1}) \prod_{t=k+2}^T p(y_t|\theta, \pi^*(h_{t-1}))} [\log p(\theta|h_T^2) - \log p(\theta)]. \end{aligned} \quad (70)$$

To extend the optimal policy to $k + 1$, we consider choosing ξ_{k+1} and then following π^* thereafter. As (70) shows us, the decision problem for ξ_{k+1} is the same starting from h_k^1 and h_k^2 because the posterior distributions $p(\theta|h_k^1)$ and $p(\theta|h_k^2)$ are equal, and the optimal policy after $k + 1$ does not depend on history order. This implies that the optimal choice of ξ_{k+1} is the same for h_k^1 and h_k^2 . This implies that the optimal policies starting in h_k^1 and h_k^2 are the same. This completes the proof. $\square$

Theorem 4. For a design function π and a number of contrastive samples $L \geq 1$, let

$$\mathcal{U}_T(\pi, L) = \mathbb{E} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} \right] \quad (71)$$

where the expectation is over $\theta_0, h_T \sim p(\theta, h_T|\pi)$ and $\theta_{1:L} \sim p(\theta)$ independently. Then,

$$\mathcal{U}_T(\pi, L) \downarrow \mathcal{I}_T(\pi) \text{ as } L \rightarrow \infty \quad (72)$$

at a rate $\mathcal{O}(L^{-1})$.

Proof. We first show $\mathcal{U}_T(\pi, L)$ is an upper bound to $\mathcal{I}_T(\pi)$

$$\mathcal{U}_T(\pi, L) - \mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} \right] - \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \quad (73)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log p(h_T|\pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi) \right) \right] \quad (74)$$

now using Jensen's inequality

$$\geq \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\log p(h_T|\pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L \mathbb{E}_{p(\theta_\ell)} [p(h_T|\theta_\ell, \pi)] \right) \right] \quad (75)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\log p(h_T|\pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L p(h_T|\pi) \right) \right] \quad (76)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} [\log p(h_T|\pi) - \log p(h_T|\pi)] \quad (77)$$

$$= 0. \quad (78)$$

To show monotonicity in L , pick $L_2 \geq L_1 \geq 0$ and consider the difference

$$\delta := \mathcal{U}_T(\pi, L_1) - \mathcal{U}_T(\pi, L_2) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{\frac{1}{L_2} \sum_{j=1}^{L_2} p(h_T|\theta_j, \pi)}{\frac{1}{L_1} \sum_{i=1}^{L_1} p(h_T|\theta_i, \pi)} \right]. \quad (79)$$

Notice that we can write expression in the numerator $\frac{1}{L_2} \sum_{j=1}^{L_2} p(h_T|\theta_j, \pi) = \mathbb{E}_{J_1, \dots, J_{L_1}} \left[\frac{1}{L_1} \sum_{k=1}^{L_1} p(h_T|\theta_{J_k}, \pi) \right]$, where the indices J_k have been uniformly drawn from $1, \dots, L_2$. We have

$$\delta = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \mathbb{E}_{J_1, \dots, J_{L_1}} \left[\frac{1}{L_1} \sum_{k=1}^{L_1} p(h_T|\theta_{J_k}, \pi) \right] - \log \frac{1}{L_1} \sum_{i=1}^{L_1} p(h_T|\theta_i, \pi) \right] \quad (80)$$

now applying Jensen's Inequality

$$\geq \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\mathbb{E}_{J_1, \dots, J_{L_1}} \left[\log \frac{1}{L_1} \sum_{k=1}^{L_1} p(h_T|\theta_{J_k}, \pi) \right] - \log \frac{1}{L_1} \sum_{i=1}^{L_1} p(h_T|\theta_i, \pi) \right] \quad (81)$$

then use the fact that any L_1 -subset of $\theta_1, \dots, \theta_{L_2}$ has the same distribution

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L_2})} \left[\log \frac{1}{L_1} \sum_{i=1}^{L_1} p(h_T|\theta_i, \pi) - \log \frac{1}{L_1} \sum_{i=1}^{L_1} p(h_T|\theta_i, \pi) \right] = 0 \quad (82)$$

which establishes monotonicity.

Finally, convergence is shown analogously to Theorem 2. Again we adopt the assumption (52). The Strong Law of Large Numbers gives us almost sure convergence $\log \left(\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi) \right) \rightarrow \log p(h_T|\pi)$ as $L \rightarrow \infty$. Applying the Bounded Convergence Theorem, as in Theorem 2, we have

$$\lim_{L \rightarrow \infty} (\mathcal{U}_T(\pi, L) - \mathcal{I}_T(\pi, L)) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\lim_{L \rightarrow \infty} \log \frac{p(h_T|\pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} \right] \quad (83)$$

$$= 0. \quad (84)$$

Finally, for the rate of convergence, we have

$$\mathcal{U}_T(\pi, L) - \mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{p(h_T|\pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_T|\theta_\ell, \pi)} \right] \quad (85)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[-\log \left(\frac{1}{L} \sum_{\ell=1}^L \frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} \right) \right] \quad (86)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[-\log \left(1 + \frac{1}{L} \sum_{\ell=1}^L \left(\frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} - 1 \right) \right) \right] \quad (87)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\sum_{n=1}^{\infty} (-1)^n \frac{x^n}{n} \right] \quad (88)$$

where $x = \frac{1}{L} \sum_{\ell=1}^L \left(\frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} - 1 \right)$ and we have applied the Taylor expansion for $\log(1+x)$. We have

$$\mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} [x] = 0 \quad (89)$$

$$\mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} [x^2] = \frac{1}{L} \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\left(\frac{p(h_T|\theta_\ell, \pi)}{p(h_T|\pi)} - 1 \right)^2 \right] \quad (90)$$

and higher order terms are $o(L^{-1})$ (Angelova, 2012; Nowozin, 2018). This shows that $\mathcal{U}_T(\pi, L) - \mathcal{I}_T(\pi) \rightarrow 0$ at a rate $\mathcal{O}(L^{-1})$. This concludes the proof. $\square$

B. Additional bounds

In this section, we consider a more general lower bound on $\mathcal{I}_T(\pi)$ based on the ACE bound of Foster et al. (2020). We consider a parametrized proposal distribution $q(\theta; h_T)$ which can be used to approximate the posterior $p(\theta|h_T)$. One example of such a proposal would be an amortized variational approximation to the posterior that takes as input h_T and outputs a variational distribution over θ . It would be possible to share the representation $R(h_T)$ from (17) between the design network and the inference network. However, the following theorem is not limited to variational posteriors, and concerns any parametrized proposal distribution.

Theorem 5. For a design function π , a number of contrastive samples $L \geq 1$, and a parametrized proposal $q(\theta; h_T)$, we have the sequential Adaptive Contrastive Estimation (sACE) lower bound

$$\mathcal{I}_T(\pi) \geq \mathbb{E}_{p(\theta_0, h_T|\pi)q(\theta_{1:L}; h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell, \pi)p(\theta_\ell)}{q(\theta_\ell; h_T)}} \right] \quad (91)$$

and the sequential Variational Nested Monte Carlo (sVNMC) upper bound

$$\mathcal{I}_T(\pi) \leq \mathbb{E}_{p(\theta_0, h_T|\pi)q(\theta_{1:L}; h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L \frac{p(h_T|\theta_\ell, \pi)p(\theta_\ell)}{q(\theta_\ell; h_T)}} \right]. \quad (92)$$

Proof. We begin by showing the sACE lower bound. The proof closely follows that of Theorem 2. We have the error term

$$\delta_{sACE} = \mathbb{E}_{p(\theta_0, h_T|\pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] - \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell, \pi)p(\theta_\ell)}{q(\theta_\ell; h_T)}} \right] \quad (93)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell, \pi)p(\theta_\ell)}{q(\theta_\ell; h_T)}}{p(h_T|\pi)} \right] \quad (94)$$

$$= \mathbb{E}_{p(\theta_0, h_T|\pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log \left(\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(\theta_\ell|h_T)}{q(\theta_\ell; h_T)} \right) \right] \quad (95)$$

now introducing the shorthand $q(\theta_{0:L}^{-\ell}; h_T) := q(\theta_{0:L \setminus \{\ell\}}; h_T) = \prod_{j=0, j \neq \ell}^L q(\theta_j; h_T)$,

$$= \mathbb{E}_{p(\theta_0, h_T | \pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell | h_T) q(\theta_{0:L}^{-\ell}; h_T)}{q(\theta_{0:L}; h_T)} \right]. \quad (96)$$

Now by the symmetry on term in side the log, we see that this expectation would be the same if it were instead taken over $p(\theta_i, h_T | \pi) q(\theta_{0:L}^{-i}; h_T)$ for any $i \in \{0, \dots, L\}$. It is also the same if we take the expectation over $\frac{1}{L+1} \sum_{i=0}^L p(\theta_i, h_T | \pi) q(\theta_{0:L}^{-i}; h_T) = p(h_T | \pi) \frac{1}{L+1} \sum_{i=0}^L p(\theta_i | h_T) q(\theta_{0:L}^{-i}; h_T)$ and thus we have

$$= \mathbb{E}_{p(h_T | \pi)} \mathbb{E}_{\frac{1}{L+1} \sum_{i=0}^L p(\theta_i | h_T) q(\theta_{0:L}^{-i}; h_T)} \left[\log \frac{\frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell | h_T) q(\theta_{0:L}^{-\ell}; h_T)}{q(\theta_{0:L}; h_T)} \right] \quad (97)$$

$$= \mathbb{E}_{p(h_T | \pi)} [\text{KL}(\check{q}(\theta_{0:L}; h_T) || q(\theta_{0:L}; h_T))] \quad (98)$$

where $\check{q}(\theta_{0:L}; h_T) = \frac{1}{L+1} \sum_{\ell=0}^L p(\theta_\ell | h_T) q(\theta_{0:L}^{-\ell}; h_T)$, which is indeed a distribution since

$$\int \check{q}(\theta_{0:L}; h_T) d\theta_{0:L} = \frac{1}{L+1} \sum_{\ell=0}^L \left(\int p(\theta_\ell | h_T) d\theta_\ell \cdot \int q(\theta_{0:L}^{-\ell}; h_T) d\theta_{0:L}^{-\ell} \right) = 1. \quad (99)$$

Now by Gibb's inequality the expected KL in (98) must be non-negative, establishing the required lower bound.

Turning to the sVNMCMC bound, we use a proof that is close in spirit to Theorem 4. We have the error term

$$\delta_{sVNMCMC} = \mathbb{E}_{p(\theta_0, h_T | \pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log \frac{p(h_T | \theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L \frac{p(h_T | \theta_\ell, \pi) p(\theta_\ell)}{q(\theta_\ell; h_T)}} \right] - \mathbb{E}_{p(\theta_0, h_T | \pi)} \left[\log \frac{p(h_T | \theta_0, \pi)}{p(h_T | \pi)} \right] \quad (100)$$

$$= \mathbb{E}_{p(\theta_0, h_T | \pi)} \mathbb{E}_{q(\theta_{1:L}; h_T)} \left[\log p(h_T | \pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L \frac{p(h_T | \theta_\ell, \pi) p(\theta_\ell)}{q(\theta_\ell; h_T)} \right) \right] \quad (101)$$

now using Jensen's inequality

$$\geq \mathbb{E}_{p(\theta_0, h_T | \pi)} \left[\log p(h_T | \pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L \mathbb{E}_{q(\theta_\ell; h_T)} \left[\frac{p(h_T | \theta_\ell, \pi) p(\theta_\ell)}{q(\theta_\ell; h_T)} \right] \right) \right] \quad (102)$$

$$= \mathbb{E}_{p(\theta_0, h_T | \pi)} \left[\log p(h_T | \pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L \mathbb{E}_{p(\theta_\ell)} [p(h_T | \theta_\ell, \pi)] \right) \right] \quad (103)$$

$$= \mathbb{E}_{p(\theta_0, h_T | \pi)} \left[\log p(h_T | \pi) - \log \left(\frac{1}{L} \sum_{\ell=1}^L p(h_T | \pi) \right) \right] \quad (104)$$

$$= \mathbb{E}_{p(\theta_0, h_T | \pi)} [\log p(h_T | \pi) - \log p(h_T | \pi)] \quad (105)$$

$$= 0. \quad (106)$$

This establishes the upper bound. $\square$

C. Gradient details

C.1. Score function gradient

Recall that our sPCE objective is

$$\mathcal{L}_T(\pi_\phi, L) = \mathbb{E}_{p(\theta_{0:L})p(h_T|\theta_0, \pi_\phi)} [g_L(\theta_{0:L}, h_T)] \quad (107)$$

$$= \mathbb{E}_{p(\theta_{0:L})p(h_T|\theta_0, \pi_\phi)} \left[\log \frac{p(h_T|\theta_0, \pi_\phi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \right] \quad (108)$$

$$= \mathbb{E}_{p(\theta_{0:L})p(h_T|\theta_0, \pi_\phi)} \left[\log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \right] + \log(L+1) \quad (109)$$

Differentiating this gives:

$$\frac{d\mathcal{L}_T}{d\phi} = \mathbb{E}_{p(\theta_{0:L})} \left[\int \frac{d}{d\phi} \left(p(h_T|\theta_0, \pi_\phi) \log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \right) dh_T \right] \quad (110)$$

$$= \mathbb{E}_{p(\theta_{0:L})} \left[\int \log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \frac{d}{d\phi} p(h_T|\theta_0, \pi_\phi) + p(h_T|\theta_0, \pi_\phi) \frac{d}{d\phi} \log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} dh_T \right] \quad (111)$$

$$= \mathbb{E}_{p(\theta_{0:L})} \left[\int p(h_T|\theta_0, \pi_\phi) \log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \left(\frac{d}{d\phi} \log p(h_T|\theta_0, \pi_\phi) \right) dh_T \right] \quad (112)$$

$$+ \int p(h_T|\theta_0, \pi_\phi) \left(\frac{d}{d\phi} \log p(h_T|\theta_0, \pi_\phi) \right) dh_T - \int p(h_T|\theta_0, \pi_\phi) \frac{d}{d\phi} \log \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi) dh_T \right] \quad (113)$$

$$= \mathbb{E}_{p(\theta_{0:L})} \mathbb{E}_{p(h_T|\theta_0, \pi_\phi)} \left[\log \frac{p(h_T|\theta_0, \pi_\phi)}{\sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \left(\frac{d}{d\phi} \log p(h_T|\theta_0, \pi_\phi) \right) - \frac{d}{d\phi} \log \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi) \right]. \quad (114)$$

In line (112) we used the log-trick $\frac{d}{dx} f(x) = f(x) \left(\frac{d}{dx} \log f(x) \right)$ and again in line (114) (in the reverse direction), together with the fact $\int \frac{d}{d\phi} p(h_T|\theta_0, \pi_\phi) dh_T = \frac{d}{d\phi} \int p(h_T|\theta_0, \pi_\phi) dh_T = 0$.

C.2. Expanded reparametrized gradient

For completeness, we provided a fully expanded form of the gradient in (14), computed using the chain rule. In practice, derivatives of this form are calculated automatically in PyTorch (Paszke et al., 2019).

Initially, we set up some additional notation. Suppose ξ the design is of dimension D_1 and y the observation is of dimension D_2 . Then $u = (\xi, y)$ is of dimension $D_1 + D_2$. For an arbitrary scalar quantity x , we have

$$\frac{\partial x}{\partial u} = \begin{pmatrix} \frac{\partial x}{\partial \xi^{(1)}} & \dots & \frac{\partial x}{\partial \xi^{(D_1)}} & \frac{\partial x}{\partial y^{(1)}} & \dots & \frac{\partial x}{\partial y^{(D_2)}} \end{pmatrix} \quad (115)$$

and

$$\frac{\partial u}{\partial x} = \begin{pmatrix} \frac{\partial \xi^{(1)}}{\partial x} & \dots & \frac{\partial \xi^{(D_1)}}{\partial x} & \sum_{d=1}^{D_1} \frac{\partial y^{(1)}}{\partial \xi^{(d)}} \frac{\partial \xi^{(d)}}{\partial x} & \dots & \sum_{d=1}^{D_1} \frac{\partial y^{(D_2)}}{\partial \xi^{(d)}} \frac{\partial \xi^{(d)}}{\partial x} \end{pmatrix}^\top. \quad (116)$$

This notation enables us to concisely and clearly deal with both scalar and vector quantities. In general, the derivatives $\partial a / \partial b$ and da / db represent a matrix of shape $(\dim a, \dim b)$ where one or both of a, b may have dimension 1. This notation is particularly attractive because the Chain Rule for partial derivatives can be concisely expressed as follows. Suppose $a = a(b_1(c), \dots, b_n(c), c)$, then the total derivative is given by

$$\frac{da}{dc} = \frac{\partial a}{\partial c} + \sum_{i=1}^n \frac{\partial a}{\partial b_i} \frac{db_i}{dc} \quad (117)$$

where the normal rules of matrix multiplication apply. We now apply this in the context of the function $g(\theta_{0:L}, h_T)$ which was defined in Section 4.2.

We have $g = g(\theta_{0:L}, u_1, \dots, u_T)$. The Chain Rule implies that

$$\frac{dg}{d\phi} = \sum_{t=1}^T \frac{\partial g}{\partial u_t} \frac{du_t}{d\phi}. \quad (118)$$

We also have, for $t = 1, \dots, T$, that $u_t = u(\phi, h_{t-1}, \theta_0, \epsilon_t) = u(\phi, u_1, \dots, u_{t-1}, \theta_0, \epsilon_t)$. This represents the dependence of ξ_t on h_{t-1} via π_ϕ , and the further dependence of y_t on θ_0 and ϵ_t . Expanding the derivatives again using the Chain Rule gives

$$\frac{dg}{d\phi} = \sum_{t=1}^T \frac{\partial g}{\partial u_t} \left(\frac{\partial u_t}{\partial \phi} + \sum_{s=1}^{t-1} \frac{\partial u_t}{\partial u_s} \frac{du_s}{d\phi} \right). \quad (119)$$

Again, we can expand the total derivative to give

$$\frac{dg}{d\phi} = \sum_{t=1}^T \frac{\partial g}{\partial u_t} \left(\frac{\partial u_t}{\partial \phi} + \sum_{s=1}^{t-1} \frac{\partial u_t}{\partial u_s} \left(\frac{\partial u_s}{\partial \phi} + \sum_{r=1}^{s-1} \frac{\partial u_s}{\partial u_r} \frac{du_r}{d\phi} \right) \right). \quad (120)$$

Rather than continuing in this manner, we observe that the current expansion (120) can be split up as follows

$$\frac{dg}{d\phi} = \sum_{t=1}^T \frac{\partial g}{\partial u_t} \frac{\partial u_t}{\partial \phi} + \sum_{1 \leq s < t \leq T} \frac{\partial g}{\partial u_t} \frac{\partial u_t}{\partial u_s} \frac{\partial u_s}{\partial \phi} + \sum_{1 \leq r < s < t \leq T} \frac{\partial g}{\partial u_t} \frac{\partial u_t}{\partial u_s} \frac{\partial u_s}{\partial u_r} \frac{du_r}{d\phi} \quad (121)$$

which shows that we have completely enumerated over all paths of length 1 and 2 through the computational graph, and the final term with a total derivative concerns paths of length 3 or more. This approach can be naturally extended to enumerate over all paths. To write this concisely, we introduce a new variable k which denotes the length of the path, and then a sum over all increasing sequences $1 \leq t_1 < \dots < t_k \leq T$. This gives

$$\frac{dg}{d\phi} = \sum_{k=1}^T \left[\sum_{1 \leq t_1 < \dots < t_k \leq T} \frac{\partial g}{\partial u_{t_k}} \frac{\partial u_{t_k}}{\partial u_{t_{k-1}}} \dots \frac{\partial u_{t_2}}{\partial u_{t_1}} \frac{\partial u_{t_1}}{\partial \phi} \right]. \quad (122)$$

This can be written concisely as

$$\frac{dg}{d\phi} = \sum_{\substack{k \in \{1, \dots, T\} \\ 1 \leq t_1 < \dots < t_k \leq T}} \frac{\partial g}{\partial u_{t_k}} \left(\prod_{j=1}^{k-1} \frac{\partial u_{t_{j+1}}}{\partial u_{t_j}} \right) \frac{\partial u_{t_1}}{\partial \phi} \quad (123)$$

where the product is interpreted in the order given in (122) for the matrix multiplication to operate correctly, and an empty product is equal to the identity.

D. Experiment details

Our experiments were implemented using PyTorch (Paszke et al., 2019) and Pyro (Bingham et al., 2018). An open-source implementation of DAD, including code for reproducing each experiment, is available at <https://github.com/ae-foster/dad>. Full details on running the code are given in the README.md file.

D.1. Location Finding

In this experiment we have K hidden objects or *sources* in $\mathbb{R}^d$, $d \in \{1, 2, 3\}$ and aim to learn their locations, $\theta = \{\theta_k\}_{k=1}^K$. The number of sources, K , is assumed to be known. Each of the sources emits a signal with intensity obeying the inverse-square law. In other words, if a source is located at θ_k and we perform a measurement at a point ξ , the signal strength will be proportional to $\frac{1}{\|\theta_k - \xi\|^2}$.

Since there are multiple sources, we consider the total intensity at location ξ , which is a superposition of the individual ones

$$\mu(\theta, \xi) = b + \sum_{k=1}^K \frac{\alpha_k}{m + \|\theta_k - \xi\|^2}, \quad (124)$$

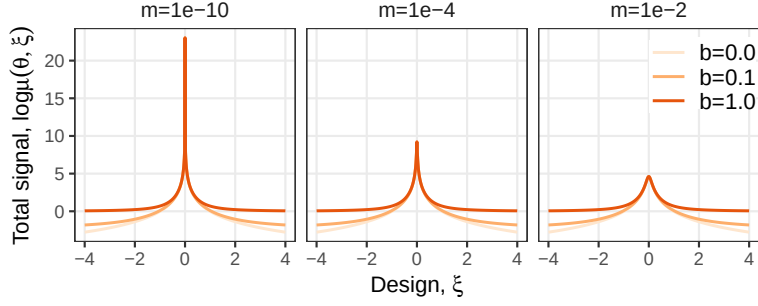


Figure 5. Log-total intensity

where α_k can be known constants or random variables, $b, m > 0$ are constants controlling background and maximum signal, respectively. Figure 5 shows the effect b and m have on log total signal strength.

We place a standard normal prior on each of the location parameters θ_k and we observe the log total intensity with some Gaussian noise. We therefore have the following prior and likelihood:

$$\theta_k \stackrel{\text{i.i.d.}}{\sim} N(0_d, I_d), \log y \mid \theta, \xi \sim N(\log \mu(\theta, \xi), \sigma). \quad (125)$$

The model hyperparameters used in our experiments can be found in the table below.

Parameter	Value
Number of sources, K	2
Base signal, b	10^{-1}
Max signal, m	10^{-4}
α_1, α_2	1
Signal noise, σ	0.5

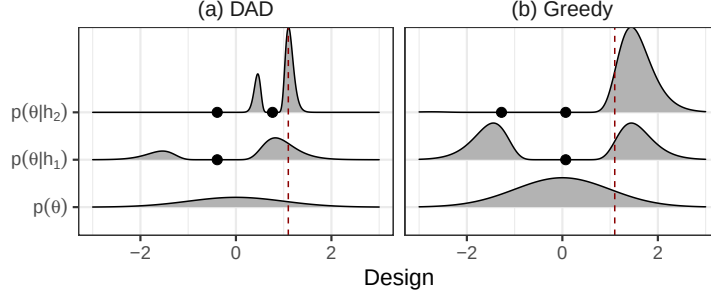
We trained a DAD network to amortize experimental design for this problem, using the neural architecture outlined in Section 4.3. Both the encoder and the decoder are simple feed-forward neural networks with a single hidden layer; details in the following table. For the encoder

Layer	Description	Dimension	Activation
Input	ξ, y	3	-
H1	Fully connected	256	ReLU
Output	Fully connected	16	-

and for the emitter

Layer	Description	Dimension	Activation
Input	$R(h_t)$	16	-
H1	Fully connected	2	-
Output	ξ	2	-

Since the likelihood is reparametrizable, we use (14) to calculate approximate gradients. We optimized the network using Adam (Kingma & Ba, 2014) with exponential learning rate annealing with parameter γ . Full details are given in the following table.


 Figure 6. Posterior distributions of the location finding example with $K = 1$ source $\mathbb{R}$.

Parameter	Value
Inner samples, L	2000
Outer samples	2000
Initial learning rate	5×10^{-5}
Betas	(0.8, 0.998)
γ	0.98
Gradient steps	50000
Annealing frequency	1000

We used a greater number of inner and outer samples for a more accurate estimate of $\mathcal{I}_T(\pi)$ for evaluation when computing the presented values in Table 1 and in our Training Stability ablation, specifically $L = 5 \times 10^5$ inner samples, and 256 (variational) or 2048 (other methods) outer samples.

Deployment times Deployment speed tests were performed on a CPU-only machine with the following specifications:

Memory	16 GB 2133 MHz LPDDR3
Processor	2.8 GHz Quad-Core Intel Core i7
Operating System	MacOS BigSur v.11.2.3

We took the mean and ± 1 s.e. over 10 realizations. Deployment times for all methods are given in the following table

Method	Deployment time (s)
Random	0.0026 ± 0.0001
Fixed	0.0018 ± 0.0001
DAD	0.0474 ± 0.0003
Variational	8963.2 ± 42.2

Discussion details In the discussion, we used a simpler form of the same model with $K = 1$ source and $\theta \in \mathbb{R}, \xi \in \mathbb{R}$. In this simplified setting, we can calculate the true optimal myopic (greedy) baseline using numerical integration. We evaluate equation (1) using line integrals as follows

$$I_t(\xi) = \int p(\theta|h_{t-1})\mathbb{E}_{p(y|\theta)} \left[\log \frac{p(y|\theta)}{\int p(\theta'|h_{t-1})p(y|\theta')d\theta'} \right] d\theta \quad (126)$$

$$= \int p(\theta|h_{t-1})\mathbb{E}_{p(y|\theta)} \left[\log \int p(\theta'|h_{t-1})p(y|\theta')d\theta' \right] d\theta + C \quad (127)$$

where $C = -H(p(y|\theta))$ is the entropy of a Gaussian, location independent and therefore constant with respect to ξ . We calculate (127) for a range of designs, $\xi \in \Xi_{\text{grid}}$, and select the optimal design $\xi^* = \arg \max_{\Xi_{\text{grid}}} I_t(\xi)$. The integrals themselves are also calculated using numerical integration on a grid, Θ_{grid} , and use sampling to calculate the inner expectation; further details can in the table below.

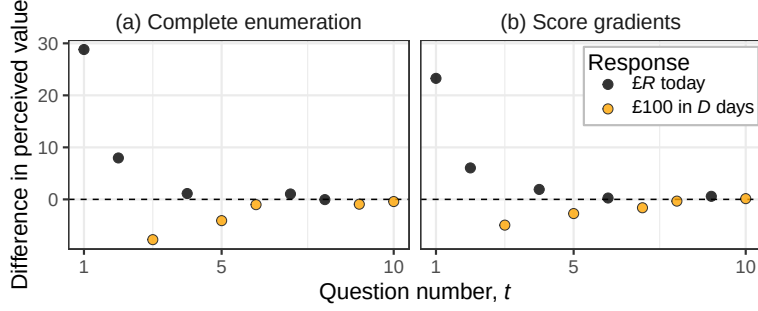


Figure 7. Comparison of two gradient methods for the hyperbolic temporal discounting model with $T = 10$ experiments.

Parameter	Value
Design grid, Ξ_{grid}	300 equally spaced from -3 to 3
θ grid, Θ_{grid}	600 equally spaced from -4 to 4
y samples for inner expectation	400

It is important to emphasize that even in this simple one-dimensional setting evaluating the myopic strategy is extremely costly and may require more sophisticated numerical integration techniques (e.g. quadrature) as posteriors become more peaked. Furthermore, as Figure 6 indicates, the resulting posteriors are complex and multi-modal even in 1D. This multi-modality may also be a reason why the variational method does not work well in this example.

D.2. Hyperbolic temporal discounting

We consider a hyperbolic temporal discounting model (Mazur, 1987; Vincent, 2016; Vincent & Rainforth, 2017) in which a participant’s behaviour is characterized by the latent variables $\theta = (k, \alpha)$ with prior distributions

$$\log k \sim N(-4.25, 1.5) \quad \alpha \sim \text{HalfNormal}(0, 2) \quad (128)$$

where the HalfNormal distribution is 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

$$V_0 = R, \quad V_1 = \frac{100}{1 + kD}. \quad (129)$$

The probability of the participant selecting the second option, V_1 , rather than V_0 is then modelled as

$$p(y = 1 | k, \alpha, R, D) = \epsilon + (1 - 2\epsilon)\Phi\left(\frac{V_1 - V_0}{\alpha}\right) \quad (130)$$

where Φ is the c.d.f. of the standard Normal distribution, i.e.

$$\Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}z^2\right) \quad (131)$$

and we fix $\epsilon = 0.01$. We considered the iterated version of this experiment, modelling $T = 20$ experiments with each sampled setting for the latents k, α .

We began by training a DAD network to amortize experimental design for this problem. The design parameters R, D have the constraints $D > 0$ and $0 < R < 100$. We represented R, D in an unconstrained space ξ_d, ξ_r and transformed them using the maps

$$D = \exp(\xi_d) \quad R = 100 \text{ sigmoid}(\xi_r) \quad (132)$$

We used the neural architecture outlined in Section 4.3. For the encoder E_{ϕ_1} we used the following network with two hidden layers

Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design

Layer	Description	Dimension	Activation
Design input	ξ_d, ξ_r	2	-
H1	Fully connected	256	Softplus
H2	Fully connected	256	Softplus
H3	Fully connected	16	-
H3'	Fully connected	16	-
Output	$y \odot H3 + (1 - y) \odot H3'$	16	-

The emitter network F_{ϕ_2} similarly used two hidden layers as follows

Layer	Description	Dimension	Activation
Input	$R(h_t)$	16	-
H1	Fully connected	256	Softplus
H2	Fully connected	256	Softplus
Output	ξ_d, ξ_r	2	-

Since the number of experiments we perform is relatively large ($T = 20$), we constructed a score function gradient estimator of (16) (see also § C.1 for details) and optimized this network with Adam (Kingma & Ba, 2014). We used exponential learning rate annealing with parameter γ . Full details are given in the following table.

Parameter	Value
Inner samples, L	500
Outer samples	500
Initial learning rate	10^{-4}
Betas	(0.9, 0.999)
γ	0.96
Gradient steps	100000
Annealing frequency	1000

For the fixed baseline, we used the same optimization settings, except we set the initial learning rate to 10^{-1} . We trained the DAD and fixed methods on a machine with 8 Intel(R) Xeon(R) CPU E5-2637 v4 @ 3.50GHz CPUs, one GeForce GTX 1080 Ti GPU, 126 GiB memory running Fedora 32. Note this is *not* the machine used to conduct speed tests. For the Badapted baseline of Vincent & Rainforth (2017), we used the public code provided at <https://github.com/drbenvincent/badapted>. We used 50 PMC steps with 100 particles. For the baselines of Frye et al. (2016) and Kirby (2009), we used the public code provided at <https://github.com/drbenvincent/darc-experiments-matlab/tree/master/darc-experiments>, which we reimplemented in Python. These methods do not involve a pre-training step, except that we did not include time to compute the first design ξ_1 within the speed test, as this can be computed before the start of the experiment.

To implement the deployment speed tests fairly, we ran each method on a lightweight CPU-only machine, which more closely mimics the computer architecture that we might expect to deploy methods such as DAD on. The specifications of the machine we used are described below

Memory	7.7GiB
Processor	Intel Core M-5Y10c CPU @ 0.80GHz $\times$ 4
Operating System	Ubuntu 16.04 LTS

The values in Table 2 show the mean and standard error of the times observed from 10 independent runs on a idle system. To make the final evaluation for each method in Table 3, we computed the sPCE and sNMC bounds using $L = 5000$ inner samples and 10000 outer samples of the outer expectation. We present the mean and standard error from the outer expectation over 10000 rollouts.

D.2.1. ABLATION: TOTAL ENUMERATION

We compare the two methods for estimating gradients for the case of discrete observations: total enumeration of histories (Equation 15) and score function gradient estimator (Equation 16). To this end we train DAD networks to perform $T = 10$ experiments, which gives rise to a total of $2^{10} = 1024$ possible histories.

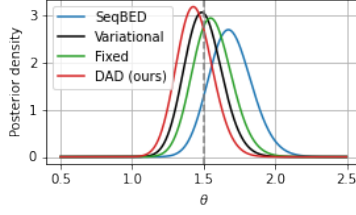


Figure 8. Comparison of posteriors obtained from a single rollout of the Death Process, used to compute the information gains quoted in Section 6.3. The dashed line indicates the true value $\theta = 1.5$ used to simulate responses.

Find that the two methods perform the same, both quantitatively and qualitatively. Table 5 reports the estimated upper and lower bounds on the mutual information objective, indicating statistically equal performance of the two methods (mean estimates are within 2 standard errors of each other). Figure 7 demonstrates the qualitative similarity in the designs learnt by the two networks.

	Lower bound, $\mathcal{L}_{10}$	Upper bound, $\mathcal{U}_{10}$
Complete enumeration	4.068 ± 0.0124	4.090 ± 0.0126
Score function gradient	4.037 ± 0.0126	4.058 ± 0.0128

Table 5. Final lower and upper bounds on the total information $\mathcal{I}_{10}(\pi)$ for the Hyperbolic Temporal Discounting experiment with $T = 10$ experiments and different gradient estimation schemes (see § 4.2 and § C.1 for details). The bounds are finite sample estimates of $\mathcal{L}_{10}(\pi, L)$ and $\mathcal{U}_{10}(\pi, L)$ with $L = 5000$. The errors indicate ± 1 s.e. over the sampled histories.

D.3. Death process

For the Death Process model (Cook et al., 2008), we use the settings that were described by Kleinegesse et al. (2020). Specifically, we use a truncated Normal prior for the infection rate

$$\theta \sim \text{TruncatedNormal}(\mu = 1, \sigma = 1, \min = 0, \max = \infty). \quad (133)$$

The likelihood is then given by

$$\eta = 1 - \exp(-\xi\theta) \quad y|\theta, \xi \sim \text{Binomial}(N, \eta) \quad (134)$$

where we set $N = 50$. We consider a sequential version of this experiment as in Kleinegesse et al. (2020), with $T = 4$ and in which an independent stochastic process is observed at each step, meaning there are no constraints relating $\xi_1, \dots, \xi_4$ other than the natural constraint $\xi_t > 0$.

We began by training a DAD network to perform experimental design for this problem. We used the neural architecture outlined in Section 4.3. For the encoder E_{ϕ_1} we used the following network with two hidden layers

Layer	Description	Dimension	Activation
Input	ξ, y	2	-
H1	Fully connected	128	Softplus
H2	Fully connected	128	Softplus
Output	Fully connected	16	-

The emitter network F_{ϕ_2} similarly used two hidden layers as follows

Layer	Description	Dimension	Activation
Input	$R(h_t)$	16	-
H1	Fully connected	128	Softplus
H2	Fully connected	128	Softplus
Output	ξ	1	Softplus

Although the number of experiments we perform is relatively small ($T = 4$), we could not use complete enumeration due to the prohibitively large size of the outcome space ($|\mathcal{Y}| = 51$). Hence, we constructed a score function gradient estimator of (16) (see also § C.1 for details) and optimized the DAD network with Adam (Kingma & Ba, 2014). We used exponential learning rate annealing with parameter γ . Full details are given in the following table.

Parameter	Value
Inner samples, L	500
Outer samples	500
Initial learning rate	0.001
Betas	(0.9, 0.999)
γ	0.96
Gradient steps	100000
Annealing frequency	1000

For the fixed baseline, we used the same optimization settings, except we set the initial learning rate to 10^{-1} and we set $\gamma = 0.85$. We trained the DAD and fixed methods using the same machine as used for training in Section D.2. For the variational baseline, we used a truncated Normal variational family to approximate the posterior at each step. We used SGD with momentum to optimize the design at each step, and to optimize the variational approximation to the posterior at each step. We used exponential learning rate annealing with parameter γ . The settings used were

Parameter	Value
Design inner samples	250
Design outer samples	250
Design initial learning rate	10^{-2}
Design γ	0.9
Design gradient steps	5000
Inference initial learning rate	10^{-3}
Inference γ	0.2
Inference gradient steps	5000
Momentum	0.1
Annealing frequency	1000

For the SeqBED baseline, we used the code publicly available at <https://github.com/stevenkleinegesse/seqbed>. The speed tests except for SeqBED were implemented as in Section D.2. For SeqBED and the variational method, we did not include the time to compute the first design as deployment time, as this can be computed before the start of the experiment. Due to its long-running nature, we implemented the speed test for SeqBED using a more powerful machine with 40 Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz processors and 189GiB memory. Therefore, the timing value for SeqBED given in Table 4 represents a significant *under-estimate* of the expected computational time required to deploy this method. However, we note that SeqBED can be applied to a broader class of implicit likelihood models.

For evaluation of $\mathcal{I}_4(\pi)$ in the Death Process, it is possible to compute the information gain $H[p(\theta)] - H[p(\theta|h_T)]$ to high accuracy using numerical integration. We then took the expectation of the information gain over rollouts, see Table 4 for the exact number of rollouts used. This gives us an estimate

$$\mathcal{I}_4(\pi) = \mathbb{E}_{p(h_T|\pi)} [H[p(\theta)] - H[p(\theta|h_T)]] \quad (135)$$

which is shown to be a valid form for the total EIG in Section A.

For a comparison with SeqBED which is too slow to use this evaluation, we instead performed one rollout of each of our methods using a fixed value $\theta = 1.5$. This is close in spirit to the evaluation used in Kleinegesse et al. (2020). Figure 8 shows the posterior distributions obtained from this rollout. The information gains were then computed using the aforementioned numerical integration and are quoted in Section 6.3. We observe that, visually, the posterior distributions are similar, and cluster near to the true value of θ .

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design
Publication Status	Published
Publication Details	Foster, A., Ivanova, D. R., Malik, I., & Rainforth, T. (2021, July). Deep adaptive design: Amortizing sequential Bayesian experimental design. In <i>International conference on machine learning</i> (pp. 3384-3395). PMLR.

Student Confirmation

Student Name:	Desi R Ivanova		
Contribution to the Paper	Co-first author. - Contributed to the development of the methodology, specifically of the second (and final) version of Sec 4.2 (Gradient estimation). Ran all of the experiments (Sec 6) with this final version of the methodology. - Contributed to the writing of the paper, mostly to experiments and proofs of the theoretical results. - Contributed significantly to the development of the codebase that was used for the empirical evaluations in the paper.		
Signature 	Date	18 April 2024	

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Dr Tom Rainforth		
Supervisor comments The contribution statement above is correct; Desi made a substantial contribution to the project.		
Signature 	Date	19/04/24

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 3

Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods

This paper is published as:

Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael U Gutmann, Thomas Rainforth. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. In *Proceedings of the Thirty-fifth Conference on Neural Information Processing Systems*. Curran Associates, Inc., 2021.

Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods

Desi R. Ivanova[†] Adam Foster[†] Steven Kleinegesse[‡] Michael U. Gutmann[‡] Tom Rainforth[†]

[†]Department of Statistics, University of Oxford

[‡]School of Informatics, University of Edinburgh

desi.ivanova@stats.ox.ac.uk

Abstract

We introduce implicit Deep Adaptive Design (iDAD), a new method for performing adaptive experiments in *real-time* with *implicit* models. iDAD amortizes the cost of Bayesian optimal experimental design (BOED) by learning a design *policy network* upfront, which can then be deployed quickly at the time of the experiment. The iDAD network can be trained on any model which simulates differentiable samples, unlike previous design policy work that requires a closed form likelihood and conditionally independent experiments. At deployment, iDAD allows design decisions to be made in milliseconds, in contrast to traditional BOED approaches that require heavy computation during the experiment itself. We illustrate the applicability of iDAD on a number of experiments, and show that it provides a fast and effective mechanism for performing adaptive design with implicit models.

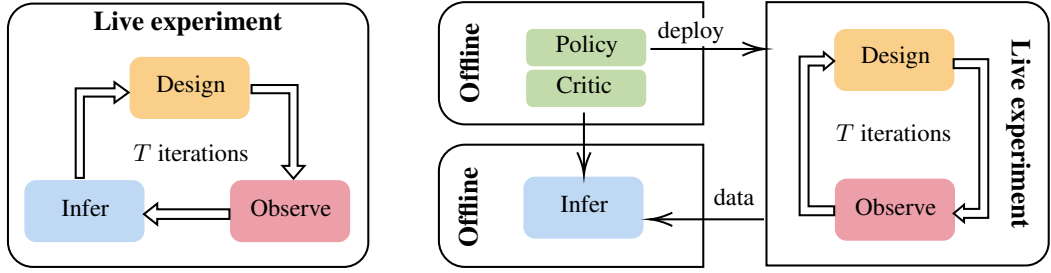
1 Introduction

Designing experiments to maximize the information gathered about an underlying process is a key challenge in science and engineering. Most such experiments are naturally *adaptive*—we can design later iterations on the basis of data already collected, refining our understanding of the process with each step [36, 45, 51]. For example, suppose that a chemical contaminant has accidentally been released and is rapidly spreading; we need to quickly discover its unknown source. To this end, we measure the contaminant concentration level at locations $\xi_1, \dots, \xi_T$ (our experimental designs), obtaining observations $y_1, \dots, y_T$. Provided we can perform the necessary computations sufficiently quickly, we can design each ξ_t using data from steps $1, \dots, t-1$ to narrow in on the source.

Bayesian optimal experimental design (BOED) [7, 32] is a principled model-based framework for choosing designs optimally; it has been successfully adopted in a diverse range of scientific fields [52, 58, 60]. In BOED, the unknown quantity of interest (e.g. contaminant location) is encapsulated by a parameter θ , and our initial information about it by a prior $p(\theta)$. A simulator, or likelihood, model $y|\theta, \xi$ describes the relationship between θ , our controllable design ξ , and the experimental outcome y . To select designs *optimally*, the guiding principle is *information maximization*—we select the design that maximizes the expected (Shannon) information gained about θ from the data y , or, equivalently, that maximizes the mutual information between θ and y .

This naturally extends to adaptive settings by considering the *conditional* expected information gain given previously collected data. The traditional approach, depicted in Figure 1a, is to fit a posterior $p(\theta|\xi_{1:t-1}, y_{1:t-1})$ after each iteration, and then select ξ_t in a myopic fashion using the one-step mutual information (see, e.g., [51] for a review). Unfortunately, this approach necessitates significant computation at each t and does not lend itself to selecting optimal designs quickly and adaptively.

Recently, Foster et al. [17] proposed an exciting alternative approach, called Deep Adaptive Design (DAD), that is based on learning design *policies*. DAD provides a way to avoid significant computation



(a) Traditional BOED: costly computations (design optimisation and parameter inference) are required at each iteration.

(b) Policy-based BOED using iDAD: a design policy and critic are learnt before the live experiment. The policy enables quick and adaptive experiments, the critic assists likelihood-free inference.

Figure 1: Overview of adaptive BOED approaches applicable to implicit models.

at deployment-time by, prior to the experiment itself, learning a design policy network that takes past design-outcome pairs and near-instantaneously returns the design for the next stage of the experiment. The required training is done using simulated experimental histories, without the need to estimate any posterior or marginal distributions. DAD further only needs a single policy network to be trained for multiple experiments, further allowing for *amortization* of the adaptive design process. Unfortunately, DAD requires conditionally independent experiments and only works for the restricted class of models that have an explicit likelihood model we can simulate from, evaluate the density of, and calculate derivatives for, substantially reducing its applicability.

To address this shortfall, we instead consider a far more general class of models where we require only the ability to simulate $y|\theta, \xi$ and compute the derivative $\partial y/\partial \xi$, e.g. via automatic differentiation [5]. Such models are ubiquitous in scientific modelling and include differentiable *implicit models* [19], for which the likelihood density $p(y|\theta, \xi)$ is intractable. Examples include mixed effects models [15, 18], various models from chemistry and epidemiology [1], the Lotka Volterra model used in ecology [19], and models specified via stochastic differential equations (such as the SIR model [10]).

To perform rapid adaptive experimentation with this large class of models, we introduce *implicit Deep Adaptive Design* (iDAD), a method for learning adaptive design policy networks using only simulated outcomes (see Figure 1b). To achieve this, we introduce likelihood-free lower bounds on the total information gained from a sequence of experiments, which iDAD utilizes to learn a deep policy network. This policy network amortizes the cost of experimental design for implicit models and can be run in milliseconds at deployment-time. To train it, we show how the InfoNCE [57] and NWJ [37] bounds, popularized in representation learning, can be applied to the policy-based experimental design setting. The optimization of both of these bounds involves simultaneously learning an auxiliary *critic network*, bringing an important added benefit: it can be used to perform likelihood-free posterior inference of the parameters given the data acquired from the experiment.

We also relax DAD’s requirement for experiments to be conditionally independent, allowing its application in complex settings like time series data, and, through innovative architecture adaptations, also provide improvements in the conditionally independent setting as well. This further expands the model space for policy-based BOED, and leads to additional performance improvements.

Critically, iDAD forms the first method in the literature that can practically perform real-time adaptive BOED with implicit models: previous approaches are either not fast enough to run in real-time for non-trivial models, or require explicit likelihood models. We illustrate the applicability of iDAD on a range of experimental design problems, highlighting its benefits over existing baselines, even finding that it often outperforms costly non-amortized approaches. Code for iDAD is publicly available at <https://github.com/desi-ivanova/idad>.

2 Background

The BOED framework [32] begins by specifying a Bayesian model of the experimental process, consisting of a prior on the unknown parameters $p(\theta)$, a set of controllable designs ξ , and a data generating process that depends on them $y|\theta, \xi$; as usual in BOED, we assume that $p(\theta)$ does not depend on ξ . In this paper, we consider the situation where $y|\theta, \xi$ is specified *implicitly*. This means

that it is defined by a deterministic transformation, $f(\varepsilon; \theta, \xi)$, of a base (or noise) random variable, ε , that is independent of the parameters and the design; e.g., $\varepsilon \sim \mathcal{N}(\varepsilon; 0, I)$. The function f is itself often not known explicitly in closed form, but is implemented as a stochastic computer program (i.e. simulator) with input (θ, ξ) and ε corresponding to the draws from the underlying random number generator (or equivalently the random seed). Regardless, the resulting induced likelihood density $p(y|\theta, \xi)$ is still generally intractable, but sampling $y|\theta, \xi$ is possible.

Having acquired a design-outcome pair (ξ, y) , we can quantify the amount of information we have gained about θ by calculating the reduction in entropy from the prior to the posterior. We can further assess the quality of a design ξ before acquiring y , by computing the expected reduction in entropy with respect to the marginal distribution of the outcome, $p(y|\xi) = \mathbb{E}_{p(\theta)}[p(y|\theta, \xi)]$. The resulting quantity, called the *expected information gain* (EIG), is of central interest in BOED and is defined as

$$I(\xi) := \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{p(\theta|\xi, y)}{p(\theta)} \right] = \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{p(y|\theta, \xi)}{p(y|\xi)} \right]. \quad (1)$$

Note that $I(\xi)$ is equivalent to the mutual information (MI) between the parameters θ and data y when performing experiment ξ . The optimal ξ is then the one that maximises the EIG, i.e. $\xi^* = \arg \max_{\xi} I(\xi)$. Performing this optimization is a major computational challenge since the information objective is doubly intractable [46]. For implicit models, the cost becomes even greater as the likelihood is also not available in closed form, so estimating it, along with the marginal likelihood $p(y|\xi)$, is already itself a major computational problem [11, 20, 33, 54].

Jointly optimizing the design variables for all undertaken experiments at the same time using (1) is called *static* experimental design. In practice, however, we are often more interested in performing multiple experiments *adaptively* in a sequence $\xi_1, \dots, \xi_T$, so that the choice of each ξ_t can be guided by past experiments, namely the corresponding *history* $h_{t-1} := \{(\xi_i, y_i)\}_{i=1:t-1}$. The typical approach in such settings is to sequentially perform (approximate) posterior inference for $\theta|h_{t-1}$, followed by a one-step look ahead (myopic) BOED optimization that conditions on the observed history. In other words, to determine the designs $\xi_1, \dots, \xi_T$, we sequentially optimize the objectives

$$I_{h_{t-1}}(\xi_t) := \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t, h_{t-1})} \left[\log \frac{p(y_t|\theta, \xi_t, h_{t-1})}{p(y_t|\xi_t, h_{t-1})} \right], \quad t = 1, \dots, T. \quad (2)$$

However, such approaches incur significant computational cost during the experiment itself, particularly for implicit models [16, 21, 30]. This has critical consequences: in most cases they cannot be run in real-time, undermining one's ability to use them in practice.

2.1 Policy-based adaptive design with likelihoods

For tractable likelihood models, Foster et al. [17] proposed a new framework, called Deep Adaptive Design (DAD), for adaptive experimental design that avoids expensive computations during the experiment. To achieve this, they introduce a parameterized deterministic design function, or policy, π_ϕ that takes the history h_{t-1} as input and returns the design $\xi_t = \pi_\phi(h_{t-1})$ to be used for the next experiment as output. This set-up allows them to consider the objective

$$\mathcal{I}_T(\pi_\phi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi_\phi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \right], \quad \xi_t = \pi_\phi(h_{t-1}), \quad (3)$$

which crucially depends on the policy π rather than the individual design ξ_t . Learning a policy up-front, rather than designs, is what allows adaptive experiments to be performed in real-time.

Under the assumption that y_t is independent of h_{t-1} conditional on the parameters θ and the design ξ_t , i.e. $p(y_t|\theta, \xi_t, h_{t-1}) = p(y_t|\theta, \xi_t)$, Foster et al. [17] showed that the objective can be simplified to

$$\mathcal{I}_T(\pi_\phi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi_\phi)} \left[\log \frac{p(h_T|\theta, \pi_\phi)}{p(h_T|\pi_\phi)} \right], \quad p(h_T|\theta, \pi_\phi) = \prod_{t=1}^T p(y_t|\theta, \xi_t). \quad (4)$$

To deal with the marginal $p(h_T|\pi_\phi)$ in the denominator, they then derived several optimizable lower bounds on $\mathcal{I}_T(\pi_\phi)$, such as the sequential Prior Contrastive Estimation (sPCE) bound

$$\mathcal{L}_T^{\text{sPCE}}(\pi_\phi, L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta, \pi_\phi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi_\phi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\theta_\ell, \pi_\phi)} \right] \leq \mathcal{I}_T(\pi_\phi) \quad \forall L \geq 1. \quad (5)$$

The parameters of the policy ϕ , which takes the form of a deep neural network, are now learned prior to the experiment(s) using stochastic gradient ascent on this bound with simulated experimental histories. Design decisions can then be made using a single forward pass of π_ϕ during deployment. Unfortunately, training the DAD network by optimizing (5) requires the likelihood density $p(h_T|\theta, \pi)$ to be analytically available—an assumption that is too restrictive in many practical situations. The architecture for DAD also assumes conditionally independent designs, which is unsuitable in some settings like time-series data. Our method lifts both of these restrictions.

3 Implicit Deep Adaptive Design

We have seen that the traditional step-by-step approach to adaptive design for implicit models [16, 21, 30] is too costly to deploy for most applications, whilst the only existing policy-based approach, DAD [17], makes restrictive assumptions that prevent it being applied to implicit models. We aim to relax the restrictive assumptions of the latter, making policy-based BOED applicable to all models where we can sample from $y|\theta, \xi$ and compute the derivative $\partial y/\partial \xi$, a strict superset of the class of models that can be handled by DAD. This requires new training objectives for the policy network that do not involve an explicit likelihood and are not based on conditionally independent experiments, along with new architectures that work for non-exchangeable models like time series.

3.1 Information lower bounds for policy-based experimental design without likelihoods

To establish a suitable likelihood-free training objective for the implicit setting, our high-level idea is to leverage recent advances in variational MI [see 42, for an overview], which have shown promise for *static* BOED [16, 28, 29]. While using these bounds in the traditional framework of (2) would not permit real-time experiments, one could consider a naive application of them to the policy objective of (3) by replacing each $I_{h_{t-1}}$ with a suitable variational lower bound that uses a ‘critic’ $U_t : \mathcal{H}^{t-1} \times \Theta \rightarrow \mathbb{R}$ to avoid explicit likelihood evaluations, where $\mathcal{H}^{t-1}$ and Θ are the spaces of histories and parameters respectively. An effective critic successfully encapsulates the true likelihood, tightening the bound. Although its form depends on the choice of bound, all critics are parametrized and trained in the same way, namely by a neural network U_{ϕ_t} which is optimized to tighten the bound. Unfortunately, replacing each $I_{h_{t-1}}$ involves learning T such critic networks and requires samples from all posteriors $p(\theta|h_{t-1})$, which will typically be impractically costly.

To avoid this issue, we show that we can obtain a unified information objective similar to (4), *even without conditionally independent experiments*. The following proposition therefore marks the first key milestone in eliminating the restrictive assumptions of [17], by establishing a unified objective without intermediate posteriors that is valid even when the model itself changes between time steps.

Proposition 1 (Generalized total expected information gain). *Consider the data generating distribution $p(h_T|\theta, \pi) = \prod_{t=1:T} p(y_t|\theta, \xi_t, h_{t-1})$, where $\xi_t = \pi(h_{t-1})$ are the designs generated by the policy and, unlike in (4), y_t is allowed to depend on the history h_{t-1} . Then we can write (3) as*

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi)] - \mathbb{E}_{p(h_T|\pi)} [\log p(h_T|\pi)]. \quad (6)$$

Proofs are presented in Appendix A. The advantage of (6) is that we can draw samples from $p(\theta)p(h_T|\theta, \pi)$ simply by sampling our model and taking forward passes through the design network. However, neither of the *densities* $p(h_T|\theta, \pi)$ and $p(h_T|\pi)$ are tractable for implicit models.

To side-step this intractability, we observe that $\mathcal{I}_T(\pi)$ takes an analogous form to a MI between θ and h_T . For measure-theoretic reasons, namely because the $\xi_{1:T}$ are deterministic given $y_{1:T}$ (see Appendix A for a full discussion), it is not the true MI. However, the following two propositions show that we can treat $\mathcal{I}_T(\pi)$ *as if it were this MI*. Specifically, we show that the InfoNCE [57] and NWJ [37] bounds on the MI can be adapted to establish tractable lower bounds on our unified objective $\mathcal{I}_T(\pi)$. These two bounds both utilize a *single* auxiliary critic network U_ψ that is trained simultaneously with the design network.

Proposition 2 (NWJ bound for implicit policy-based BOED). *For a design policy π and a critic function $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$, let*

$$\mathcal{L}_T^{\text{NWJ}}(\pi, U) := \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta)] - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} [\exp(U(h_T, \theta))], \quad (7)$$

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{\text{NWJ}}(\pi, U)$ holds for any U . Further, the inequality is tight for the optimal critic $U_{\text{NWJ}}^*(h_T, \theta) = \log p(h_T|\theta, \pi) - \log p(h_T|\pi) + 1$.

Algorithm 1: Implicit Deep Adaptive Design with (iDAD)

Input: Differentiable simulator f , sampler for prior $p(\theta)$, number of experimental steps T

Output: Design network π_ϕ , critic network U_ψ

while Computational training budget not exceeded **do**

 Sample $\theta \sim p(\theta)$ and set $h_0 = \emptyset$

for $t = 1, \dots, T$ **do**

 Compute $\xi_t = \pi_\phi(h_{t-1})$

 Sample $\varepsilon_t \sim p(\varepsilon)$ and compute $y_t = f(\varepsilon_t; \xi_t, \theta, h_{t-1})$

 Set $h_t = \{(\xi_1, y_1), \dots, (\xi_t, y_t)\}$

end

 Estimate $\nabla_{\phi, \psi} \mathcal{L}_T(\pi_\phi, U_\psi)$ as per (10) where $\mathcal{L}_T$ is $\mathcal{L}_T^{\text{NWJ}}$ (7) or $\mathcal{L}_T^{\text{NCE}}$ (8)

 Update the parameters (ϕ, ψ) using stochastic gradient ascent scheme

end

For deployment, use the deterministic trained design network π_ϕ to obtain a designs ξ_t directly.

Proposition 3 (InfoNCE bound for implicit policy-based BOED). *Let $\theta_{1:L} \sim p(\theta_{1:L}) = \prod_i p(\theta_i)$ be a set of contrastive samples where $L \geq 1$. For design policy π and critic function $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$, let*

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

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{\text{NCE}}(\pi, U; L)$ for any U and $L \geq 1$. Further, the optimal critic, $U_{\text{NCE}}^*(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$ where $c(h_T)$ is any arbitrary function depending only on the history, recovers the sPCE bound in (5); the inequality is tight in the limit as $L \rightarrow \infty$ for this optimal critic.

We propose these two alternative bounds due to their complementary properties: the NWJ bound can have large variance, but tends to be less biased. That is, the NWJ bound tends to be tighter for good critics, but is itself more difficult to reliably estimate and thus optimize. While the NWJ critic must learn to self-normalize, the InfoNCE bound avoids this issue but typically will not be tight for finite L even with an optimal critic (note $\mathcal{L}_T^{\text{NCE}} \leq \log(L+1)$ [42]). Consequently, only the NWJ objective recovers the true optimal policy if our critic has infinite capacity and our optimization scheme is perfect, i.e. $\arg \max_\pi \max_U \mathcal{L}_T^{\text{NWJ}}(\pi, U) = \pi^* \neq \arg \max_\pi \max_U \mathcal{L}_T^{\text{NCE}}(\pi, U; L)$ in general, but it can be more difficult to work with in practice. We present a third bound that provides a potential solution to this, and further discuss the relative merits of the two bounds, in Appendix A.

We note that for both bounds the optimal critic does not depend on the learned policy. The final trained critic can be used to approximate the density ratio $p(h_T|\theta, \pi)/p(h_T|\pi) = p(\theta|h_T)/p(\theta)$, either directly in the case of the NWJ critic, or via self-normalization for the InfoNCE bound. We can use this to help approximate the posterior over θ given the collected real data from the experiment. This means we can perform likelihood-free inference after training the critic, which extends previous results [28, 29] from the static to the adaptive policy-based setting.

3.2 Parameterization and gradient estimation

In practice, we represent the policy π and the critic U as neural networks, π_ϕ and U_ψ respectively, such that the lower bounds become a function $\mathcal{L}(\pi_\phi, U_\psi)$ of their parameters. By simultaneously optimizing $\mathcal{L}(\pi_\phi, U_\psi)$ with respect to both ϕ and ψ , we both learn a tight bound that accurately represents the true MI and a design policy network that produces high-quality designs under this metric.

We optimize these bounds using stochastic gradient methods [26, 49]. For this, we must account for the fact that the parameter ϕ affects the probability distributions with respect to which expectations are taken. We deal with this problem by utilizing the reparametrization trick [35, 48], for which we assume that design space Ξ and observation space $\mathcal{Y}$ are continuous. To this end, we first formalize the notion of a differentiable implicit model in the adaptive design setting as

$$y_t = f(\varepsilon_t; \xi_t(h_{t-1}), \theta, h_{t-1}), \quad \text{where } \theta \sim p(\theta), \quad \varepsilon_t \sim p(\varepsilon) \quad \forall t \in \{1, \dots, T\} \quad (9)$$

and we assume that we can compute the derivatives $\partial f / \partial \xi$ and $\partial f / \partial h$. Interestingly, it is possible to use an implicit prior without access to the density $p(\theta)$, and we do not need access to $\partial f / \partial \theta$.

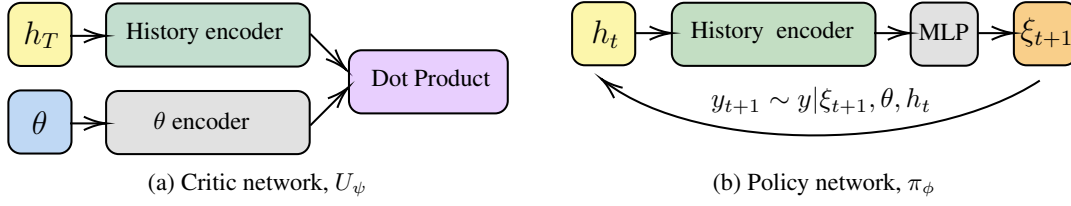


Figure 2: Overview of network architectures used in iDAD.

Under these conditions, we can express the bounds in terms of expectations that do not depend on ϕ or ψ , and hence move the gradient operator inside. For $\mathcal{L}_T^{\text{NCE}}(\pi_\phi, U_\psi; L)$, for example, we have

$$\nabla_{\phi, \psi} \mathcal{L}_T^{\text{NCE}} = \mathbb{E}_{p(\theta_{0:L})p(\varepsilon_{1:T})} \left[\nabla_{\phi, \psi} \log \frac{\exp(U_\psi(h_T(\varepsilon_{1:T}, \pi_\phi), \theta_0))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U_\psi(h_T(\varepsilon_{1:T}, \pi_\phi), \theta_i))} \right]. \quad (10)$$

While each element of the history h_T depends on ϕ in a possibly nested manner, we do not need to explicitly keep track of these dependencies thanks to automatic differentiation [5, 41].

Like DAD, our new method—which we call *implicit Deep Adaptive Design* (iDAD)—is trained with simulated histories $h_T = \{(\xi_i, y_i)\}_{i=1:T}$ prior to the actual experiment, allowing design decision to be made using a single forward pass during deployment. Unlike DAD, however, it does not require knowledge of the likelihood function, nor the assumption of conditionally independent designs, which significantly broadens its applicability. A summary of the iDAD approach is given in Algorithm 1.

3.3 Network architectures

The iDAD approach involves the simultaneous training of the *policy* π_ϕ and *critic* U_ψ networks. It is essential to choose the neural architectures of these two components carefully to learn effective policies: poor choices of critic architecture will lead to loose, unrepresentative, bounds, while poor choices of policy architecture will directly lead to ineffective policies. Good choices of architecture need to balance flexibility with ease of training, and will typically require the incorporation of problem-specific inductive biases. A high-level summary of our architectures is shown in Figure 2.

The critic network, U_ψ , takes a *complete* history h_T and the parameter θ as input, and outputs a scalar. Our suggested architecture first encodes the two inputs separately to representations of the same dimension, using a *history encoder*, E_{ψ_h} , and a *parameter encoder*, E_{ψ_θ} , respectively. The output of the critic is then simply taken as their dot product $U_\psi(h_T, \theta) := E_{\psi_h}(h_T)^\top E_{\psi_\theta}(\theta)$; after training, the two encodings correspond to approximate sufficient statistics [9]. This setup corresponds to a separable critic architecture, as is commonly used in the representation learning literature [3, 8, 57]. While we use a simple MLP for E_{ψ_θ} , the setup for E_{ψ_h} varies with the context as we discuss below.

The policy network, π_ϕ , takes the available history, h_t , as input, and outputs a design. Our suggested architecture makes use of a history encoder, E_{ϕ_h} , of the same form as E_{ψ_h} , except that it must now take in varying length inputs; its output remains a fixed dimensional embedding. We then pass this embedding through an MLP to produce the design ξ_{t+1} . At the next iteration of the experiment, the same policy network is then called again with the updated history $h_{t+1} = h_t \cup \{(\xi_{t+1}, y_{t+1})\}$.

We use the same architecture for both history encoders, E_{ψ_h} and E_{ϕ_h} , but do not share network parameters between them. This architecture first individually embeds each design–outcome pair (ξ_t, y_t) to a corresponding representation, r_t , using a simple MLP that is shared across all time steps. The produced history encoding is then an aggregation of these representations, with how this is done depending on whether the experiments are *conditionally independent*, i.e. $y_t \perp\!\!\!\perp h_{t-1} | \theta, \xi_t$, or not.

Foster et al. [17] proved that if experiments are conditionally independent, then the optimal policy is invariant to the order of the history. We prove that the same is true for the critic in Proposition 5 in Appendix C. In our setup, we can exploit this result by using a *permutation invariant* aggregation strategy for $\{r_1, \dots, r_t\}$ when conditional independence holds. The simplest approach to do this would be to use sum-pooling [62], as was done in DAD. However, to improve on this, we instead propose using a more advanced permutation invariant architecture based on self-attention [13, 25, 39, 47, 59], namely that of Parmar et al. [40]; we find this provides notable empirical gains. When conditional independence does not hold, this approach is no longer appropriate and we instead use an LSTM [22] for the aggregation. See Appendix C for further details.

4 Related work

Adaptive policy-based BOED has only recently been introduced [17] and has not yet been extended to implicit models—the gap that this work addresses. Previous approaches to adaptive experiments usually follow the two-step greedy procedure described in Section 2. Methods for MI/EIG estimation without likelihoods include the use of variational bounds [15, 16, 28] and ratio estimation [27, 30]; approximate Bayesian computation together with kernel density estimation [43]; and approximating the intractable likelihood first, for example via polynomial chaos expansion [24], followed by applying likelihood-based estimators, such as nested Monte Carlo [46]. The maximization step in more traditional methods tends to rely on gradient-free optimization, including grid-search, evolutionary algorithms [44], Bayesian optimization [15, 30], or Gaussian process surrogates [38]. More recently, gradient-based approaches have been introduced [15, 28], some of which allow the estimation and optimization simultaneously in a single stochastic-gradient scheme [16, 23, 29]. From a posterior estimation perspective, likelihood-free inference can be performed via approximate Bayesian computation [33, 54], ratio estimation [56], conventional MCMC for methods that make tractable approximation to the likelihood [23, 24], or as a byproduct of MI estimation [16, 27, 29, 30].

5 Experiments

We evaluate the performance of **iDAD** on a number of real-world experimental design problems and a range of baselines. A summary of all the methods that we consider is given in Table 1. Since we aim to perform adaptive experiments in *real-time*, we focus mostly on baselines that do not require significant computational time during the experiment. These include heuristic approaches that require no training, namely **equal** interval designs (when possible) and **random** designs, as well as static BOED approaches, where we, non-adaptively, choose all the designs prior to the experiment by optimising the mutual information objective of Equation (1) with $\xi = \{\xi_1, \dots, \xi_T\}$ and $y = \{y_1, \dots, y_T\}$. The static BOED approaches we consider are the **MINEBED** method of [28] and the likelihood-free ACE approach of [16], where we use the prior as a proposal distribution, referring to this baseline as **SG-BOED**. We also implement the expensive traditional non-amortized myopic strategy described in Section 2, for which we use the **variational** approach of [16], with the Barber-Agakov bound [4, 15], at each experiment step (see Appendix D.3 for details). Finally, where possible, we compare our method with DAD [17], in order to assess the performance gap that would arise if we had an analytic likelihood. This comparison is done primarily for evaluation purposes—because it has access to the likelihood density, DAD serves as an upper bound on the performance iDAD can achieve; one should use explicit likelihood methods whenever possible.

Table 1: Key properties of considered methods.

	Adaptive	Real-time	Implicit
Random	✗	N/A	✓
Equal interval	✗	N/A	✓
MINEBED	✗	N/A	✓
SG-BOED	✗	N/A	✓
Variational	✓	✗	✓
DAD	✓	✓	✗
iDAD	✓	✓	✓

The main performance metric that we focus on is the total EIG, $\mathcal{I}_T(\pi)$, as given in (6). In cases where the likelihood is available, we estimate the $\mathcal{I}_T(\pi)$ using the sPCE lower bound in (5) and its sister upper bound, the sequential Nested Monte Carlo bound [sNMC; 17]. To ensure that the bounds are tight, we evaluate them with a large number of contrastive samples, i.e. $L \geq 10^5$. Where the likelihood is truly intractable, we assess the iDAD strategy in a more qualitative manner by looking at the optimal designs and approximate posteriors. For the adaptive experiments, we further consider the deployment time (i.e. the time required to propose a design), which is a critical metric for our aims. All deployment times exclude the time needed to determine the first experiment as it can be computed up-front, during the training phase. Timings for training the policy itself are given in Appendix D.

The main performance metric that we focus on is the total EIG, $\mathcal{I}_T(\pi)$, as given in (6). In cases where the likelihood is available, we estimate the $\mathcal{I}_T(\pi)$ using the sPCE lower bound in (5) and its sister upper bound, the sequential Nested Monte Carlo bound [sNMC; 17]. To ensure that the bounds are tight, we evaluate them with a large number of contrastive samples, i.e. $L \geq 10^5$. Where the likelihood is truly intractable, we assess the iDAD strategy in a more qualitative manner by looking at the optimal designs and approximate posteriors. For the adaptive experiments, we further consider the deployment time (i.e. the time required to propose a design), which is a critical metric for our aims. All deployment times exclude the time needed to determine the first experiment as it can be computed up-front, during the training phase. Timings for training the policy itself are given in Appendix D.

5.1 Location Finding

We first demonstrate our approach on the location finding experiment from [17]. Inspired by the acoustic energy attenuation model [53], this experiment involves finding the locations of multiple hidden sources, each emitting a signal with intensity that decreases according to the inverse-square law. The *total intensity*—a superposition of these signals—can be measured noisily at any location. The design problem is choosing where to measure the total signal in order to uncover the sources.

Table 2: Lower bounds on the total information, $\mathcal{I}_{10}(\pi)$, for the location finding experiment in Section 5.1. The bounds were estimated using $L = 5 \times 10^5$ contrastive samples. Errors indicate ± 1 standard errors estimated over 4096 histories (128 for variational). Corresponding upper bounds are given in Table 6 in Appendix D.

Method \ θ dim.	4D	6D	10D	20D
Random	4.791 ± 0.040	3.468 ± 0.014	1.889 ± 0.011	0.552 ± 0.006
MINEBED	5.518 ± 0.028	4.221 ± 0.028	2.458 ± 0.029	0.801 ± 0.019
SG-BOED	5.547 ± 0.028	4.215 ± 0.030	2.454 ± 0.029	0.803 ± 0.019
Variational	4.639 ± 0.144	3.625 ± 0.165	2.181 ± 0.151	0.669 ± 0.097
iDAD (NWJ)	7.694 ± 0.045	5.765 ± 0.036	3.252 ± 0.039	0.877 ± 0.022
iDAD (InfoNCE)	7.750 ± 0.039	5.986 ± 0.037	3.251 ± 0.039	0.871 ± 0.020
DAD	7.967 ± 0.034	6.300 ± 0.030	3.337 ± 0.039	0.937 ± 0.022

Table 3: Lower and upper bounds on MI $\mathcal{I}_{10}(\pi)$ for different network architectures on location finding experiment using the InfoNCE bound. All estimates obtained as in Table 2.

Design	Critic	Lower bound	Upper bound
Attention	Attention	7.750 ± 0.039	7.863 ± 0.043
Attention	Pooling	7.567 ± 0.037	7.632 ± 0.039
Pooling	Attention	7.398 ± 0.040	7.470 ± 0.042
Pooling	Pooling	7.135 ± 0.034	7.192 ± 0.041

In Table 2 we can see that iDAD substantially outperforms all baselines including, perhaps surprisingly, the traditional (non-amortized) adaptive variational approach, despite its large computational cost shown Table 4. The poor performance of the variational approach is likely driven by the inability of the mean-field variational family to capture the highly non-Gaussian true posterior, highlighting the detrimental effect that wrong posteriors can have on determining optimal designs when using the traditional sequential BOED approach.

Table 2 further shows that the performance gap to the likelihood-based DAD method is small, even as the dimension of the design and parameter space grows. Though the information gained by all methods decreases with the dimensionality, this is to be expected: in higher dimensions it is inherently more difficult to infer the relative direction of the sources from observing their intensity. Overall, this experiment demonstrates that iDAD is able to learn near-optimal amortized design policies without likelihoods, while being run in milliseconds at deployment.

Ablation: attention to history. We next assess the benefit of utilizing our proposed more sophisticated permutation invariant architectures, compared to the simple pooling of [62] used in [17]. Our approach incorporates attention layers into both networks that we train. This leads us to four possible combinations of network architectures. Table 3 compares the efficacy of the resulting design policies and strongly suggests that incorporating attention mechanisms in either and/or both networks improves performance, with inclusion in the design network particularly important.

We preform further ablation studies to investigate and demonstrate important properties of our method, such as its scalability with the number of experiments T , stability between different training runs and performance to errors in the design network (introduced by not training the network to convergence). Results and discussion are provided in Appendix D.4.4.

5.2 Pharmacokinetic model

Our next experiment is taken from the pharmacokinetics literature and has been studied in other recent works on BOED for implicit models [28, 63]. Specifically, we consider the compartmental model

Table 4: Deployment time of adaptive methods in 2D, measured on a CPU. Errors were calculated on the basis of 10 runs.

Method	Deployment time (sec.)
Variational	$2256.0 \pm 1\%$
iDAD (NWJ)	$0.0167 \pm 2\%$
iDAD (InfoNCE)	$0.0168 \pm 2\%$
DAD	$0.0070 \pm 6\%$

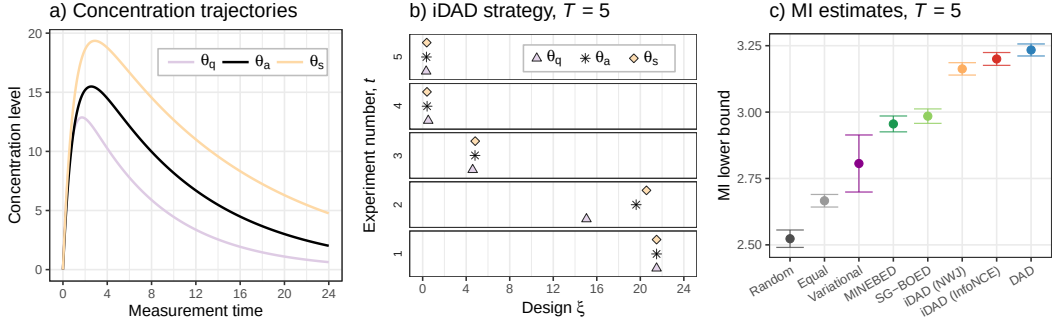


Figure 3: Plots for pharmacokinetics experiment. a) Visualisation of model showing concentration level as a function of measurement time for 3 values of θ , resulting in a quick (θ_q), average (θ_a), or slow (θ_s) trajectory. b) Designs selected by an iDAD policy trained with InfoNCE. c) MI lower bounds achieved by iDAD and baselines. All estimates obtained as in Table 2.

of [50], for which the distribution of an administered drug through the body is governed by three parameters: absorption rate k_α , elimination rate k_e , and volume V , which form the parameters of interest, i.e. $\theta = (k_\alpha, k_e, V)$. Given $T = 5$ patients, the design problem is to adaptively choose blood sampling times, $0 \leq \xi_t \leq 24$ hours, for each, measured from the the point the drug was administered (with patient 2 not being administered until after sampling patient 1 etc). Plausible concentration trajectories are shown in Figure 3a). Full details and further results are given in Appendix D.5.

We first qualitatively consider the design policy of iDAD (trained with the InfoNCE objective) in Figure 3b). As we have not yet observed any data, the optimal design for the first patient (bottom row) is the same for all θ . For the second patient, only guided by ξ_1 and the outcome y_1 , iDAD is already able to distinguish between quickly and slowly decaying concentration trajectories: it proposes a significantly earlier measurement time for the quickly decaying trajectory (purple triangle, θ_q) and later time for the slowly decaying one (yellow diamond, θ_s). For the third patient, iDAD always targets the peak of the drug concentration distribution which is quite similar for all θ . Measurements for the last two patients are made soon after the drug has been administered ($\sim 15 - 30$ min), when concentration levels increase rapidly, to capture information about how quickly the drug is absorbed.

To provide more quantitative assessment and compare to our baselines, we again consider the final EIG values as shown in Figure 3c). This reveals that the iDAD strategies perform best among the methods that are applicable to implicit models, confirming that the learnt policies propose superior designs. The performance gap to DAD, which relies on explicit likelihoods, is not statistically significant (at the 5% level) for iDAD trained with InfoNCE, while significant, but still small, for NWJ.

Finally, we consider the convergence of the iDAD networks under the different training objectives and compare to DAD for reference. As shown in Figure 4, although all three converge to approximately the same value, they do so at rather different speeds: while DAD requires about 5000 gradient updates, implicit methods need longer training and tend to exhibit higher variance, particularly NWJ.

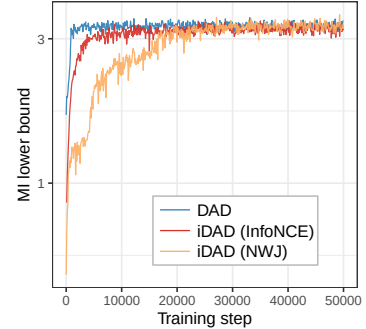


Figure 4: Convergence of MI lower bounds.

5.3 SIR Model

In this experiment, we demonstrate our approach on an implicit model from epidemiology. Namely, we consider a formulation of the stochastic SIR model [10] that is based on stochastic differential equations (SDEs), as done by [29]. Here, individuals in a fixed population belong to one of three categories: susceptible, infected or recovered. Susceptible people can become infected and then recover, with the dynamics of these two events being governed by two model parameters—the infection rate β and the recovery rate γ . Our aim is to determine the optimal times τ at which to measure the number of infected people, $I(\tau)$, in order to estimate the two parameters. This implicit model is challenging because data simulation is expensive, since we need to solve many SDEs, and experimental designs have a time-dependency. See Appendix D.6 for full details.

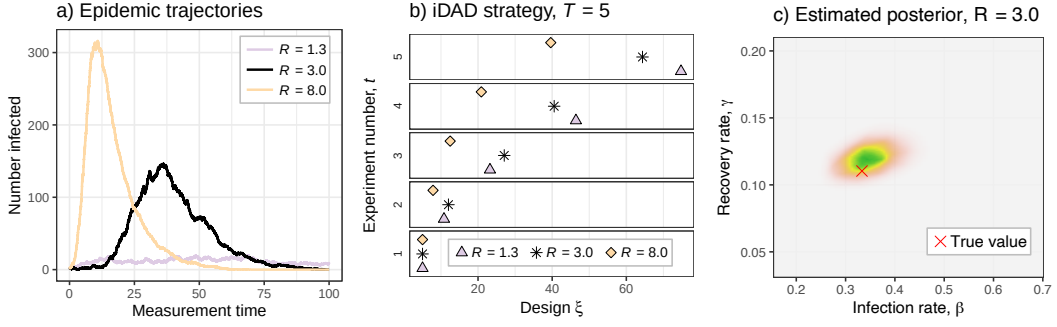


Figure 5: a) Epidemic trajectories for 3 realization of (β, γ) with different reproduction numbers $R = \beta/\gamma$. b) Designs selected by an iDAD policy trained with NWJ. c) Example posterior estimates from the critic network given data generated with the ground-truth parameters shown by the red cross.

We train a iDAD networks to perform $T = 5$ experiments and compare against random, equal interval, and static design baselines; DAD cannot be run because the problem corresponds to a true implicit model. Table 5 shows lower bound estimates on the MI and demonstrates that iDAD outperforms all compared methods. Note that a degree of caution is required when analysing the results, as they are influenced by unavoidable biases in the estimation process. Namely, a critic is still required to estimate the MI lower bound, and there may be variations in the effectiveness of these critics, with less effective ones corresponding to looser bounds and therefore underestimating the true MI. Nonetheless, for other models where such checks are possible, we have found the bounds to be relatively tight, while, even if this turns out not to be the case here, the fact that the critics for the static approaches are easier to train should mean our relative evaluations for iDAD (and Random) are still conservative compared to the other baselines.

Table 5: MI lower bounds (± 1 s.e.).

Method	Lower bound
Random	1.915 ± 0.032
Equal interval	2.669 ± 0.023
MINEBED	3.400 ± 0.001
SG-BOED	3.752 ± 0.020
iDAD (NWJ)	3.869 ± 0.001
iDAD (InfoNCE)	3.915 ± 0.020

Figure 5 further demonstrates important qualitative results for this model. Figure 5a) shows different epidemic trajectories, i.e. the number of infected $I(\tau)$ people as a function of measurement time τ , whilst 5b) plots their corresponding designs obtained from the learned iDAD policy. Importantly, diseases with a significantly different profile, e.g. a slow ($R = 1.3$) or a fast ($R = 8.0$) spread result in different sets of optimal designs, highlighting the adaptivity of iDAD. Finally, Figure 5c) shows an example posterior distribution estimate from the learnt iDAD critic network, which we see is consistent with the ground truth parameters.

6 Discussion

Limitations. The benefit that iDAD can be used in live experiments comes at the cost of substantial training that can be computationally expensive. However, this is mitigated by its amortization of the adaptive design process, such that only one network needs training, even if we have multiple experiment instances. The cost–performance trade-off can also be directly controlled by judicious choices of architecture and the amount of training performed. Another natural limitation is that the use of gradients naturally restricts the approach to continuous design settings, something which future work might look to address.

Conclusions. In this paper we introduced iDAD—the first policy-based adaptive BOED method that can be applied to implicit models. By training a design network without likelihoods upfront, iDAD is thus the first method that allows real-time adaptive experiments for simulator-based models. In our experiments, iDAD performed significantly better than all likelihood-free baselines. Further, by using models where the likelihood is available as a test bed, we found that it was able to almost match the analogous likelihood-based adaptive approach, which acts as an upper bound on what might be achieved without access to the likelihood itself. In conclusion, we believe iDAD marks a step change in Bayesian experimental design for *implicit* models, allowing designs to be proposed quickly, adaptively, and non-myopically during the live experiment.

Acknowledgments and Disclosure of Funding

DRI is supported by EPSRC through the Modern Statistics and Statistical Machine Learning (StatML) CDT programme, grant no. EP/S023151/1. AF gratefully acknowledges funding from EPSRC grant no. EP/N509711/1. SK was supported in part by the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1) and the University of Edinburgh.

References

- [1] Edward J. Allen, Linda J. S. Allen, Armando Arciniega, and Priscilla E. Greenwood. Construction of equivalent stochastic differential equation models. *Stochastic Analysis and Applications*, 26(2):274–297, 2008.
- [2] Linda J.S. Allen. A primer on stochastic epidemic models: Formulation, numerical simulation, and analysis. *Infectious Disease Modelling*, 2(2):128–142, 2017. ISSN 2468-0427. doi: <https://doi.org/10.1016/j.idm.2017.03.001>.
- [3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019.
- [4] David Barber and Felix Agakov. The im algorithm: A variational approach to information maximization. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS’03, page 201–208, Cambridge, MA, USA, 2003. MIT Press.
- [5] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18, 2018.
- [6] 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.
- [7] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.
- [8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.
- [9] Yanzhi Chen, Dinghuai Zhang, Michael U. Gutmann, Aaron Courville, and Zhanxing Zhu. Neural approximate sufficient statistics for implicit models. In *International Conference on Learning Representations*, 2021.
- [10] Alex R Cook, Gavin J Gibson, and Christopher A Gilligan. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008.
- [11] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 2020.
- [12] Mahasen B. Dehideniya, Christopher C. Drovandi, and James M. McGree. Optimal bayesian design for discriminating between models with intractable likelihoods in epidemiology. *Computational Statistics & Data Analysis*, 124:277–297, 2018. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2018.03.004>.
- [13] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 4171–4186, 2019.
- [14] Yann Dubois, Jonathan Gordon, and Andrew YK Foong. Neural Process Family. <http://yanndubs.github.io/Neural-Process-Family/>, September 2020.

- [15] 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.
- [16] 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.
- [17] 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.
- [18] Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.
- [19] Matthew Graham and Amos Storkey. Asymptotically exact inference in differentiable generative models. In *Artificial Intelligence and Statistics*, pages 499–508. PMLR, 2017.
- [20] PeterJ. Green, Krzysztof Latuszynski, Marcelo Pereyra, and Christian P. Robert. Bayesian computation: a summary of the current state, and samples backwards and forwards. *Statistics and Computing*, 25(4):835–862, 2015. doi: 10.1007/s11222-015-9574-5.
- [21] Markus Hainy, Christopher C. Drovandi, and James M. McGree. Likelihood-free extensions for Bayesian sequentially designed experiments. In Joachim Kunert, Christine H. Müller, and Anthony C. Atkinson, editors, *mODa 11 - Advances in Model-Oriented Design and Analysis*, pages 153–161. Springer International Publishing, 2016.
- [22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.
- [23] Xun Huan and Youssef Marzouk. Gradient-based stochastic optimization methods in bayesian experimental design. *International Journal for Uncertainty Quantification*, 4(6), 2014.
- [24] Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013.
- [25] Cheng Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer: Generating music with long-term structure, 2019. ISSN 23318422.
- [26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [27] 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.
- [28] 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.
- [29] 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.
- [30] 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.
- [31] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

- [32] Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956.
- [33] 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.
- [34] David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 875–884. PMLR, 2020.
- [35] 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.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] Emilio Parisotto, Francis Song, Jack Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant Jayakumar, Max Jaderberg, Raphaël Lopez Kaufman, Aidan Clark, Seb Noury, Matthew Botvinick, Nicolas Heess, and Raia Hadsell. Stabilizing transformers for reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7487–7498. PMLR, 2020.
- [40] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018.
- [41] 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.
- [42] Ben Poole, Sherjil Ozair, Aäron van den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180, 2019.
- [43] David J. Price, Nigel G. Bean, Joshua V. Ross, and Jonathan Tuke. On the efficient determination of optimal bayesian experimental designs using abc: A case study in optimal observation of epidemics. *Journal of Statistical Planning and Inference*, 172:1–15, May 2016.
- [44] 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.
- [45] Tom Rainforth. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, University of Oxford, 2017.
- [46] 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.

- [47] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jon Shlens. Stand-alone self-attention in vision models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- [48] 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.
- [49] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.
- [50] Elizabeth G. Ryan, Christopher C. Drovandi, M. Helen Thompson, and Anthony N. Pettitt. Towards bayesian experimental design for nonlinear models that require a large number of sampling times. *Computational Statistics & Data Analysis*, 70:45–60, 2014. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2013.08.017>.
- [51] 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.
- [52] Ben Shababo, Brooks Paige, Ari Pakman, and Liam Paninski. Bayesian inference and online experimental design for mapping neural microcircuits. In *Advances in Neural Information Processing Systems*, pages 1304–1312, 2013.
- [53] 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.
- [54] 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.
- [55] Jiaming Song and Stefano Ermon. Understanding the limitations of variational mutual information estimators. In *International Conference on Learning Representations*, 2020.
- [56] Owen Thomas, Ritabrata Dutta, Jukka Corander, Samuel Kaski, and Michael U Gutmann. Likelihood-free inference by ratio estimation. *arXiv preprint arXiv:1611.10242*, 2016.
- [57] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [58] Joep Vanlier, Christian A Tiemann, Peter AJ Hilbers, and Natal AW van Riel. A Bayesian approach to targeted experiment design. *Bioinformatics*, 28(8):1136–1142, 2012.
- [59] 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.
- [60] 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.
- [61] 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.
- [62] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, 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.
- [63] Jiaxin Zhang, Sirui Bi, and Guannan Zhang. A stochastic approximate gradient ascent method for Bayesian experimental design with implicit models. In *The 24th International Conference on Artificial Intelligence and Statistics*, 2021.

A Proofs

We present proof for all propositions made in the paper, restating each for convenience. We also include additional discussion on technical aspects of the paper.

A.1 Unified objective for non-exchangeable experiments

Proposition 1 (Generalized total expected information gain). *Consider the data generating distribution $p(h_T|\theta, \pi) = \prod_{t=1:T} p(y_t|\theta, \xi_t, h_{t-1})$, where $\xi_t = \pi(h_{t-1})$ are the designs generated by the policy and, unlike in (4), y_t is allowed to depend on the history h_{t-1} . Then we can write (3) as*

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi)] - \mathbb{E}_{p(h_T|\pi)} [\log p(h_T|\pi)]. \quad (6)$$

Proof. Starting with the definition of the total EIG (3) of a policy π :

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\sum_{t=1}^T I_{h_{t-1}}(\xi_t) \right] \quad (11)$$

we have by linearity of expectation

$$= \sum_{t=1}^T \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [I_{h_{t-1}}(\xi_t)] \quad (12)$$

and since $I_{h_{t-1}}$ doesn't depend on data acquired after $t-1$ (the future doesn't influence the past)

$$= \sum_{t=1}^T \mathbb{E}_{p(\theta)p(h_{t-1}|\theta, \pi)} [I_{h_{t-1}}(\xi_t)] \quad (13)$$

which, applying Bayes rule, is equivalent to

$$= \sum_{t=1}^T \mathbb{E}_{p(h_{t-1}|\pi)p(\theta|h_{t-1})} [I_{h_{t-1}}(\xi_t)] \quad (14)$$

Next, using Bayes rule we similarly rearrange $I_{h_{t-1}}$:

$$I_{h_{t-1}}(\xi_t) = \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t, h_{t-1})} \left[\log \frac{p(y_t|\theta, \xi_t, h_{t-1})}{p(y_t|\xi_t, h_{t-1})} \right] \quad (15)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t, h_{t-1})} \left[\log \frac{p(\theta|y_t, \xi_t, h_{t-1})}{p(\theta|h_{t-1})} \right] \quad (16)$$

$$= \mathbb{E}_{p(\theta|h_{t-1})p(y_t|\theta, \xi_t, h_{t-1})} [\log p(\theta|y_t, \xi_t, h_{t-1})] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \quad (17)$$

$$= \mathbb{E}_{p(\theta|y_t, \xi_t, h_{t-1})p(y_t|\xi_t, h_{t-1})} [\log p(\theta|y_t, \xi_t, h_{t-1})] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \quad (18)$$

and noting $h_t = h_{t-1} \cup \{(\xi_t, y_t)\}$

$$= \mathbb{E}_{p(\theta|h_t)p(y_t|\xi_t, h_{t-1})} [\log p(\theta|h_t)] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \quad (19)$$

$$= \mathbb{E}_{p(y_t|\xi_t, h_{t-1})} \left[\mathbb{E}_{p(\theta|h_t)} [\log p(\theta|h_t)] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \right] \quad (20)$$

Substituting this in (14), noting that θ has already been integrated out, yields

$$\mathcal{I}_T(\pi) = \sum_{t=1}^T \mathbb{E}_{p(h_{t-1}|\pi)} \mathbb{E}_{p(y_t|\xi_t, h_{t-1})} \left[\mathbb{E}_{p(\theta|h_t)} [\log p(\theta|h_t)] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \right] \quad (21)$$

$$= \sum_{t=1}^T \mathbb{E}_{p(h_t|\pi)} \left[\mathbb{E}_{p(\theta|h_t)} [\log p(\theta|h_t)] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \right] \quad (22)$$

$$= \mathbb{E}_{p(h_T|\pi)} \left[\sum_{t=1}^T \mathbb{E}_{p(\theta|h_t)} [\log p(\theta|h_t)] - \mathbb{E}_{p(\theta|h_{t-1})} [\log p(\theta|h_{t-1})] \right], \quad (23)$$

since we have a telescopic sum this simplifies to

$$= \mathbb{E}_{p(h_T|\pi)} \left[\mathbb{E}_{p(\theta|h_T)} [\log p(\theta|h_T)] - \mathbb{E}_{p(\theta)} [\log p(\theta)] \right] \quad (24)$$

and finally we apply Bayes rule again to rewrite as

$$= \mathbb{E}_{p(h_T|\pi)p(\theta|h_T)} \left[\log p(\theta|h_T) - \mathbb{E}_{p(\theta)} [\log p(\theta)] \right] \quad (25)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(\theta|h_T) - \log p(\theta)] \quad (26)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log p(h_T|\theta, \pi) - p(h_T|\pi)] \quad (27)$$

□

A.2 Objective function as a mutual information

We provide some additional discussion on the interpretation of $\mathcal{I}_T(\pi)$ in (6) as a mutual information. First, $\mathcal{I}_T(\pi)$ is not a conventional mutual information between θ and h_T . This is because, for the deterministic policy π considered in this paper, the random variable h_T does not have a density with respect to Lebesgue measure on $\Xi^T \times \mathcal{Y}^T$. Indeed, since the designs $\xi_{1:T}$ are deterministic functions of the observations $y_{1:T}$, to express the sampling distribution of h_T we would have to use Dirac deltas, specifically

$$p(y_{1:T}, \xi_{1:T} | \theta, \pi) = \prod_{t=1}^T \delta_{\pi(h_{t-1})}(\xi_t) p(y_t | \theta, \xi_t, h_{t-1}). \quad (28)$$

Due to the presence of Dirac deltas, this is not a conventional probability density, and hence we do not regard $\mathcal{I}_T(\pi)$ as the conventional mutual information between θ and h_T .

We note that we *defined* $p(h_T | \theta, \pi)$ in Proposition 1 differently to $p(y_{1:T}, \xi_{1:T} | \theta, \pi)$ in (28). Specifically, our definition

$$p(h_T | \theta, \pi) = \prod_{t=1}^T p(y_t | \theta, \xi_t, h_{t-1}) \quad (29)$$

only involves probability densities for $y_{1:T}$, meaning that our $p(h_T | \theta, \pi)$ is a well-defined probability density on $\mathcal{Y}^T$. Formally, we can treat the designs ξ_t , not as additional random variables, but as part of the density for $y_{1:T}$. Indeed, since the policy π is deterministic, it is possible to reconstruct h_{t-1} and ξ_t from $y_{1:t-1}$ and π , so we could write $p(y_t | \theta, y_{1:t-1}, \pi) := p(y_t | \theta, \xi_t, h_{t-1})$. In this formulation, only $y_{1:T}$ are regarded as random variables. This provides a formal justification for the form of $p(h_T | \theta, \pi)$ that we give in Proposition 1. In this setting, we could formally identify $\mathcal{I}_T(\pi)$ as the mutual information between θ and $y_{1:T}$.

However, it is helpful to think of $\mathcal{I}_T(\pi)$ as a mutual information between θ and h_T , because this naturally leads to critics that have access to θ and h_T , rather than θ and $y_{1:T}$. This way of thinking also connects naturally to the case of stochastic policies, which we now discuss.

If we consider additional noise in the design process so that designs are no longer a deterministic function of past data, then $\mathcal{I}_T(\pi)$ is the mutual information between θ and h_T . In this case, we introduce an additional likelihood for designs $p(\xi | \pi, h)$, leading to the overall sampling distribution for the data

$$p(h_T | \theta, \pi) = \prod_{t=1}^T p(\xi_t | \pi, h_{t-1}) p(y_t | \theta, \xi_t, h_{t-1}). \quad (30)$$

Unlike in the deterministic case, this is valid probability density on $\Xi^T \times \mathcal{Y}^T$. If we now consider the mutual information between θ and h_T for a fixed policy π we have

$$I(\theta, h_T) = \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)} \left[\log \frac{\prod_{t=1}^T p(\xi_t | \pi, h_{t-1}) p(y_t | \theta, \xi_t, h_{t-1})}{\int_{\Theta} p(\theta) \prod_{t=1}^T p(\xi_t | \pi, h_{t-1}) p(y_t | \theta, \xi_t, h_{t-1}) d\theta} \right] \quad (31)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)} \left[\log \frac{\prod_{t=1}^T p(\xi_t | \pi, h_{t-1}) \prod_{t=1}^T p(y_t | \theta, \xi_t, h_{t-1})}{\prod_{t=1}^T p(\xi_t | \pi, h_{t-1}) \int_{\Theta} p(\theta) \prod_{t=1}^T p(y_t | \theta, \xi_t, h_{t-1}) d\theta} \right] \quad (32)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta,\pi)} \left[\log \frac{\prod_{t=1}^T p(y_t | \theta, \xi_t, h_{t-1})}{\int_{\Theta} p(\theta) \prod_{t=1}^T p(y_t | \theta, \xi_t, h_{t-1}) d\theta} \right] \quad (33)$$

noticing that the design likelihood terms cancel out in the integrand, and we reduce to the same integrand given in Proposition 1. Even when the policy is stochastic, the integrand in $I(\theta, h_T)$ only involves terms of the form $p(y_t | \theta, \xi_t, h_{t-1})$, and the likelihood of the design process completely cancels. Thus, the stochasticity of the designs is only present in the sampling distribution $p(h_T | \theta, \pi)$. We therefore see that, as we consider the limiting case of $p(\xi | \pi, h)$ as it approaches a deterministic policy, only the sampling distribution of designs in $I(\theta, h_T)$ changes, with the integrand remaining the same. Under mild assumptions, then, the mutual information between θ and h_T approaches $\mathcal{I}_T(\pi)$ in this limit.

A.3 NWJ and InfoNCE bounds

The next two propositions show that the two bounds—NWJ and InfoNCE—can be applied to the policy-based adaptive BOED setting.

Proposition 2 (NWJ bound for implicit policy-based BOED). *For a design policy π and a critic function $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$, let*

$$\mathcal{L}_T^{NWJ}(\pi, U) := \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta)] - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} [\exp(U(h_T, \theta))], \quad (7)$$

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{NWJ}(\pi, U)$ holds for any U . Further, the inequality is tight for the optimal critic $U_{NWJ}^*(h_T, \theta) = \log p(h_T|\theta, \pi) - \log p(h_T|\pi) + 1$.

Proof. Let $\pi : \mathcal{H}^* \rightarrow \Xi$ be any (deterministic) policy taking histories h_t as inputs and returning a design ξ as output, $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$ be any function and define $g(h_T, \theta) := \frac{\exp(U(h_T, \theta))}{\mathbb{E}_{p(h_T|\pi)}[\exp(U(h_T, \theta))]}$.

First, we multiply the numerator and denominator of the unified objective (6) by $g(h_T, \theta) > 0$

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \right] \quad (34)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \log \left[\frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \frac{g(h_T, \theta)}{g(h_T, \theta)} \right] \quad (35)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log g(h_T, \theta)] + \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)g(h_T, \theta)} \right] \quad (36)$$

Next, note that the second term is a KL divergence between two distributions

$$\mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)g(h_T, \theta)} \right] = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(\theta)p(h_T|\theta, \pi)}{p(\theta)p(h_T|\pi)g(h_T, \theta)} \right] \quad (37)$$

$$= KL(p(\theta)p(h_T|\theta, \pi) || \hat{p}(h_T, \theta)) \geq 0 \quad (38)$$

where $\hat{p}(h_T, \theta) = p(\theta)p(h_T|\pi)g(h_T, \theta)$ is a valid distribution since

$$\int p(\theta)p(h_T|\pi)g(h_T, \theta)d\theta dh_T = \mathbb{E}_{p(\theta)p(h_T|\pi)} \frac{\exp(U(h_T, \theta))}{\mathbb{E}_{p(h_T|\pi)}[\exp(U(h_T, \theta))]} \quad (39)$$

$$= \mathbb{E}_{p(\theta)} 1 = 1. \quad (40)$$

Therefore, we have

$$\mathcal{I}_T(\pi) \geq \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [\log g(h_T, \theta)] \quad (41)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta) - \log \mathbb{E}_{p(h_T|\pi)} \exp(U(h_T, \theta))] \quad (42)$$

$$= \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta)] - \mathbb{E}_{p(\theta)} [\log \mathbb{E}_{p(h_T|\pi)} \exp(U(h_T, \theta))] \quad (43)$$

Now using the inequality $\log x \leq e^{-1}x$

$$\geq \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta)] - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} [\exp(U(h_T, \theta))] \quad (44)$$

$$= \mathcal{L}_T^{NWJ}(\pi, U) \quad (45)$$

Finally, substituting $U^*(h_T, \theta) = \log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} + 1$ in the bound we get

$$\mathcal{L}_T^{NWJ}(\pi, U^*) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} + 1 \right] - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} \left[\frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} e^1 \right] \quad (46)$$

$$= \mathcal{I}_T(\pi) + 1 - \mathbb{E}_{p(\theta)p(h_T|\pi)} \left[\frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \right] \quad (47)$$

$$= \mathcal{I}_T(\pi), \quad (48)$$

where we used $\mathbb{E}_{p(\theta)p(h_T|\pi)} \left[\frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \right] = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [1] = 1$, establishing that the bound is tight for the optimal critic.

□

Proposition 3 (InfoNCE bound for implicit policy-based BOED). *Let $\theta_{1:L} \sim p(\theta_{1:L}) = \prod_i p(\theta_i)$ be a set of contrastive samples where $L \geq 1$. For design policy π and critic function $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$, let*

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

then $\mathcal{I}_T(\pi) \geq \mathcal{L}_T^{NCE}(\pi, U; L)$ for any U and $L \geq 1$. Further, the optimal critic, $U_{NCE}^*(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$ where $c(h_T)$ is any arbitrary function depending only on the history, recovers the sPCE bound in (5); the inequality is tight in the limit as $L \rightarrow \infty$ for this optimal critic.

Proof. Let $\pi : \mathcal{H}^* \rightarrow \Xi$ be any (deterministic) policy taking histories h_t as inputs and returning a design ξ as output. Choose any function (critic) $U : \mathcal{H}^T \times \Theta \rightarrow \mathbb{R}$.

We introduce the shorthand

$$g(h_T, \theta_{0:L}) := \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U(h_T, \theta_i))} \quad (49)$$

Starting with the definition of the unified objective from Equation (6) we multiply its numerator and denominator by $g(h_T, \theta_{0:L}) > 0$ to get

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \quad (50)$$

where $p(\theta_0)p(h_T|\theta_0, \pi) \equiv p(\theta)p(h_T|\theta, \pi)$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \quad (51)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)g(h_T, \theta_{0:L})}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \quad (52)$$

We next split the expectation into two terms one of which does not contain the unknown likelihoods and equals $\mathcal{L}^{NCE}$

$$\begin{aligned} &= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \\ &\quad + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} [\log g(h_T, \theta_{0:L})] \\ &= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] + \mathcal{L}^{NCE}(\pi, U; L) \end{aligned} \quad (53)$$

We now show that the first term is a KL divergence and hence non-negative. To see why, first write

$$\mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \quad (54)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})}{p(\theta_0)p(h_T|\pi)p(\theta_{1:L})g(h_T, \theta_{0:L})} \right] \quad (55)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\log \frac{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})}{\hat{p}(\theta_{0:L}, h_T|\pi)} \right] \quad (56)$$

$$= KL(p(h_T|\theta_0, \pi)p(\theta_{0:L}) || \hat{p}(\theta_{0:L}, h_T|\pi)). \quad (57)$$

and $\hat{p}(\theta_{0:L}, h_T|\pi)$ is a valid distribution since

$$\int \hat{p}(\theta_{0:L}, h_T|\pi) d\theta_{0:L} dh_T = \int p(\theta_0)p(h_T|\pi)p(\theta_{1:L})g(h_T, \theta_{0:L}) d\theta_{0:L} dh_T \quad (58)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)p(\theta_{1:L})} \left[\frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U(h_T, \theta_i))} \right], \quad (59)$$

because of the symmetry $\theta_0 \stackrel{d}{=} \theta_j \forall j = 1, \dots, L$

$$= \frac{1}{L+1} \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{\sum_{j=0}^L \exp(U(h_T, \theta_j))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U(h_T, \theta_i))} \right] \quad (60)$$

$$= 1. \quad (61)$$

Thus we have established

$$\mathcal{I}_T(\pi) = KL(p(h_T|\theta_0, \pi)p(\theta_{0:L})||\hat{p}(\theta_{0:L}, h_T|\pi)) + \mathcal{L}_T^{NCE}(\pi, U; L) \geq \mathcal{L}_T^{NCE}(\pi, U; L). \quad (62)$$

Next, substituting $U^*(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$ in the definition of $\mathcal{L}^{NCE}(\pi, U; L)$ we obtain

$$\mathcal{L}_T^{NCE}(\pi, U^*; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{p(h_T|\theta_0, \pi) \exp(c(h_T))}{\frac{1}{L+1} \sum_{i=0}^L p(h_T|\theta_i, \pi) \exp(c(h_T))} \right] \quad (63)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L})} \left[\frac{p(h_T|\theta_0, \pi)}{\frac{1}{L+1} \sum_{i=0}^L p(h_T|\theta_i, \pi)} \right], \quad (64)$$

which is exactly the sPCE bound (5), which is monotonically increasing in L and tight in the limit as $L \rightarrow \infty$ [see 17, Theorem 2]. $\square$

A.4 A note on optimal critics

An interesting feature of our approach is that, for both the InfoNCE and NWJ bounds, the optimal critics do not depend on the policy. This is because we include the designs as explicit inputs to the critics. Indeed, we have

$$U_{\text{NCE}}^*(h_T, \theta) = \log \left(\prod_{t=1}^T p(y_t|\theta, \xi_t, h_{t-1}) \right) + c(h_T), \quad (65)$$

$$U_{\text{NWJ}}^*(h_T, \theta) = \log \left(\frac{\prod_{t=1}^T p(y_t|\theta, \xi_t, h_{t-1})}{\int_{\Theta} p(\theta) \prod_{t=1}^T p(y_t|\theta, \xi_t, h_{t-1}) d\theta} \right) + 1. \quad (66)$$

In previous work that utilized critics for gradient-based BOED [16, 28], it was typical to not treat the designs $\xi_{1:T}$ as an input to the critic, which renders the optimal critic implicitly dependent on the designs. This makes more sense for static designs, for which the additional design input does not change. Our approach avoids an implicit dependence between policy and optimal critic which may be beneficial for the joint optimization.

B Theoretical Comparison and Additional Bounds

Recently, a number of studies have discussed the challenges of estimating mutual information, in particular those associated with variational MI estimators [34, 42, 55].

Starting with the InfoNCE bound, it is trivial to show that the bound cannot exceed $\log(L+1)$, where L is the number of contrastive samples used to approximate the marginal in the denominator. Indeed,

$$\mathcal{L}_T^{NCE}(\pi, U; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U(h_T, \theta_i))} \right] \quad (67)$$

$$\leq \log(L+1) + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \mathbb{E}_{p(\theta_{1:L})} \left[\log \frac{\exp(U(h_T, \theta_0))}{\exp(U(h_T, \theta_0))} \right] \quad (68)$$

$$= \log(L+1) \quad (69)$$

This means that the corresponding Monte Carlo estimator will be highly biased whenever the true mutual information exceeds $\log(L+1)$, regardless of whether we have access to the optimal critic or not. This high bias estimator, however, comes with low variance [see e.g. 42, for discussion]. With

the optimal critic we would require exponential (in the MI) number of samples to accurately estimate the true mutual information.

It might appear at first that the NWJ bound might offer a better trade-off between bias and variance. Recall from the proof of Proposition 2, we have for the *optimal* critic

$$\mathcal{L}_T^{NWJ}(\pi, U^*) = \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} \left[\log \frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} \right] + 1 - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} \left[\frac{p(h_T|\theta, \pi)}{p(h_T|\pi)} e^1 \right], \quad (70)$$

of which we form a Monte carlo estimate using N (M) samples for the first (second) term, respectively

$$\approx \frac{1}{N} \sum_{n=1}^N \log \frac{p(h_{T,n}|\theta_n, \pi)}{p(h_{T,n}|\pi)} + \left(1 - \frac{1}{M} \sum_{m=1}^M \log \frac{p(h_{T,m}|\theta_m, \pi)}{p(h_{T,m}|\pi)} \right), \quad (71)$$

where $\theta_n, h_{T,n} \sim p(\theta)p(h_T|\theta, \pi)$ are samples from the joint distribution and $\theta_m, h_{T,m} \sim p(\theta)p(h_T|\pi)$ are samples from the product of marginals. The first term is a Monte Carlo estimate of the mutual information, while the second has mean zero, meaning that this estimator is unbiased. The second term, however has variance which grows exponentially with the value of the (true) mutual information [see Theorem 2 in 55]. What this means is that even with an optimal critic, we will need an exponential (in the MI) number of samples to control the variance of the NWJ estimator. One might then hope that the variance can be reduced when using a sub-optimal critic at the cost of introducing some (hopefully small) bias. Unfortunately, according to a recent result [see Theorems 3.1 and 4.1 in 34, and the discussion therein], it is not possible to guarantee that a likelihood-free lower bound on the mutual information can exceed $\log(N)$. Indeed, the authors show theoretically and empirically that all high-confidence distribution-free lower bounds on the mutual information require exponential (in the the MI) number of samples.

Constructing a better lower bound on the mutual information—one that does not need exponential number of samples—therefore, requires us to make additional assumptions. Foster et al. [17] propose one such bound, namely the sequential Adaptive Contrastive Estimation (sACE). The sACE bound introduces a proposal distribution $q(\theta; h_T)$, which aims to approximate the posterior $p(\theta|h_T)$. Since implicit models were not the focus of the work in [17] the proposed bound, relies on analytically available likelihood. The following proposition shows we can derive a likelihood-free version of the sACE bound.

Proposition 4 (Sequential Likelihood-free ACE). *For a design function π , a critic function U , a number of contrastive samples $L \geq 1$, and a proposal $q(\theta; h_T)$, we have the sequential Likelihood-free Adaptive Contrastive Estimation (sLACE) lower bound*

$$\mathcal{L}_T^{sLACE}(\pi, U, q; L) := \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)q(\theta_{1:L}; h_T)} \left[\log \frac{U(h_T, \theta_0)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell; h_T)}} \right] \leq \mathcal{I}_T(\pi). \quad (72)$$

The bound is tight as $L \rightarrow \infty$ for the optimal critic $U^*(h_T, \theta) = \log p(h_T|\theta, \pi) + c(h_T)$, where $c(h_T)$ is arbitrary. In addition, if $q(\theta; h_T) = p(\theta|h_T)$, the bound is tight for the optimal critic $U^*(h_T, \theta)$ with any $L \geq 0$.

Proof. The proof follows similar arguments to the ones in Propositions 2 and 3. First let

$$g(h_T, \theta_{0:L}) := \frac{U(h_T, \theta_0)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell; h_T)}} \quad (73)$$

Starting with the definition of the EIG:

$$\mathcal{I}_T(\pi) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \quad (74)$$

since $q(\theta_i; h_T)$ is a valid density

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)q(\theta_{1:L}; h_T)} \left[\log \frac{p(h_T|\theta_0, \pi)}{p(h_T|\pi)} \right] \quad (75)$$

multiplying its numerator and denominator inside the log by $g(h_T, \theta_{0:L}) > 0$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)g(h_T, \theta_{0:L})}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \quad (76)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} [\log g(h_T, \theta_{0:L})] \\ + \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \quad (77)$$

The first term is exactly the sLACE bound, $\mathcal{L}_T^{\text{sLACE}}(\pi, U, q; L)$. We now show that the second term is a KL divergence between two distributions and hence non-negative. To see this

$$\mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)}{p(h_T|\pi)g(h_T, \theta_{0:L})} \right] \quad (78)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)}{p(h_T|\pi)g(h_T, \theta_{0:L})p(\theta_0)q(\theta_{1:L};h_T)} \right] \quad (79)$$

$$= KL(p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T) \parallel \hat{p}(h_T, \theta_{0:L})), \quad (80)$$

since $\hat{p}(h_T, \theta_{0:L}) := p(h_T|\pi)g(h_T, \theta_{0:L})p(\theta_0)q(\theta_{1:L};h_T)$ is a valid density. Indeed:

$$\int \hat{p}(h_T, \theta_{0:L}) dh_T d\theta_{0:L} = \mathbb{E}_{q(\theta_{1:L};h_T)p(h_T|\pi)} [p(\theta_0)g(h_T, \theta_{0:L})] \quad (81)$$

$$= \mathbb{E}_{q(\theta_{1:L};h_T)p(h_T|\pi)} \left[p(\theta_0) \frac{U(h_T, \theta_0)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}} \right] \quad (82)$$

$$= \mathbb{E}_{q(\theta_{0:L};h_T)p(h_T|\pi)} \left[\frac{\frac{U(h_T, \theta_0)p(\theta_0)}{q(\theta_0;h_T)}}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}} \right] \quad (83)$$

by symmetry

$$= \mathbb{E}_{q(\theta_{0:L};h_T)p(h_T|\pi)} \left[\frac{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{U(h_T, \theta_\ell)p(\theta_\ell)}{q(\theta_\ell;h_T)}} \right] \quad (84)$$

$$= 1. \quad (85)$$

With the optimal critic we recover the sACE bound from [17], which under mild conditions converges to the mutual information $\mathcal{I}_T(\pi)$. To see that start by writing

$$\mathcal{L}_T^{\text{sLACE}}(\pi, U^*, q; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)q(\theta_{1:L};h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell,\pi)p(\theta_\ell)}{q(\theta_\ell;h_T)}} \right]. \quad (86)$$

The denominator is a consistent estimator of the marginal, provided that each term in the sum is bounded, and so by the Strong Law of Large Numbers we have

$$\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell,\pi)p(\theta_\ell)}{q(\theta_\ell;h_T)} \rightarrow p(h_T|\pi) \text{ a.s. as } L \rightarrow \infty, \quad (87)$$

which establishes point-wise convergence of the integrand to $p(h_T|\theta_0,\pi)/p(h_T|\pi)$. We can apply Bounded convergence theorem to establish $\mathcal{L}_T^{\text{sACE}}(\pi, U^*, q; L) \rightarrow \mathcal{I}_T(\pi)$ as $L \rightarrow \infty$.

If in addition $q(\theta; h_T) = p(\theta|h_T)$ we have by Bayes rule:

$$\mathcal{L}_T^{\text{sLACE}}(\pi, U^*, q; L) = \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L}|h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)}{\frac{1}{L+1} \sum_{\ell=0}^L \frac{p(h_T|\theta_\ell,\pi)p(\theta_\ell)}{p(\theta_\ell|h_T)}} \right] \quad (88)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)p(\theta_{1:L}|h_T)} \left[\log \frac{p(h_T|\theta_0,\pi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_T|\pi)} \right] \quad (89)$$

$$= \mathbb{E}_{p(\theta_0)p(h_T|\theta_0,\pi)} \left[\log \frac{p(h_T|\theta_0,\pi)}{p(h_T|\pi)} \right] \quad (90)$$

$$= \mathcal{I}_T(\pi) \quad \forall L \geq 0. \quad (91)$$

□

In practice, we parameterize the policy, the critic and the density of the proposal distribution by neural networks π_ϕ , U_ψ and q_ζ and optimize $\mathcal{L}_T^{\text{sLACE}}$ with respect to the parameters of these networks, ϕ , ψ and ζ with SGA. As before, optimizing with respect to ϕ improves the quality of the designs, proposed by the policy, whilst optimizing with respect to ψ and ζ tightens the bound. If the parametric density q_ζ and the critic U_ψ are expressive enough, so that we can recover the optimal critic and the true posterior, then the bound is tight for any number of contrastive samples L . If, on the other hand, we fix $q_\zeta(\theta; h_T) = p(\theta)$ instead of training it, then we recover the InfoNCE bound. Therefore, as long as q_ζ approximates the posterior better than the prior, then even an imperfect proposal q_ζ can benefit training.

In addition to introducing another set of optimizable parameters, ζ , the sLACE bound assumes that we know the prior $p(\theta)$ and can evaluate its density.

C Neural architecture

C.1 Permutation invariance of the critic for exchangeable experiments

We show that if the BOED problem is exchangeable then the critic function U should be permutation-invariant.

Proposition 5 (Permutation invariance). *Let σ be a permutation acting on a history h_T^1 yielding $h_T^2 = \{(\xi_{\sigma(i)}, y_{\sigma(i)})\}_{i=1}^T$. If the data generating process is conditionally independent of its past given θ , then the optimal critics for both (7) and (8) are invariant under permutations of the history, i.e.*

$$p(\theta) \prod_{t=1}^T p(y_t | \theta, \xi_t(h_{t-1}), h_{t-1}) = p(\theta) \prod_{t=1}^T p(y_t | \theta, \xi_t) \implies U^*(h_T^1, \theta) = U^*(h_T^2, \theta). \quad (92)$$

Proof. This is a direct consequence from the form of the optimal critics. To see this formally, let h_T^1 be a history and h_T^2 be a permutation of it.

Starting with the InfoNCE bound we have

$$U_{\text{NCE}}^*(h_T^1, \theta) = \log p(h_T^1 | \theta, \pi) + c(h_T^1) \quad (93)$$

$$= \log \prod_{t=1}^T p(y_t | \theta, \xi_t) + c(\{(\xi_t, y_t)\}_{t=1}^T) \quad (94)$$

since $c(h_T)$ is arbitrary, we can choose it to be permutation invariant

$$= \log \prod_{t=1}^T p(y_{\sigma(t)} | \theta, \xi_{\sigma(t)}) + c(\{(\xi_{\sigma(t)}, y_{\sigma(t)})\}_{t=1}^T) \quad (95)$$

$$= \log p(h_T^2 | \theta, \pi) + c(h_T^2) \quad (96)$$

$$= U_{\text{NCE}}^*(h_T^2, \theta) \quad (97)$$

Similarly, for the optimal critic of the NWJ bound we have

$$U_{\text{NWJ}}^*(h_T^1, \theta) = \log \frac{p(h_T^1 | \theta, \pi)}{p(h_T^1 | \pi)} + 1 \quad (98)$$

$$= \log \frac{\prod_{t=1}^T p(y_t | \theta, \xi_t)}{\mathbb{E}_{p(\theta)} \left[\prod_{s=1}^T p(y_s | \theta, \xi_s) \right]} + 1 \quad (99)$$

$$= \log \frac{\prod_{t=1}^T p(y_{\sigma(t)} | \theta, \xi_{\sigma(t)})}{\mathbb{E}_{p(\theta)} \left[\prod_{s=1}^T p(y_{\sigma(s)} | \theta, \xi_{\sigma(s)}) \right]} + 1 \quad (100)$$

$$= \log \frac{p(h_T^2 | \theta, \pi)}{p(h_T^2 | \pi)} + 1 = U_{\text{NWJ}}^*(h_T^2, \theta). \quad (101)$$

□

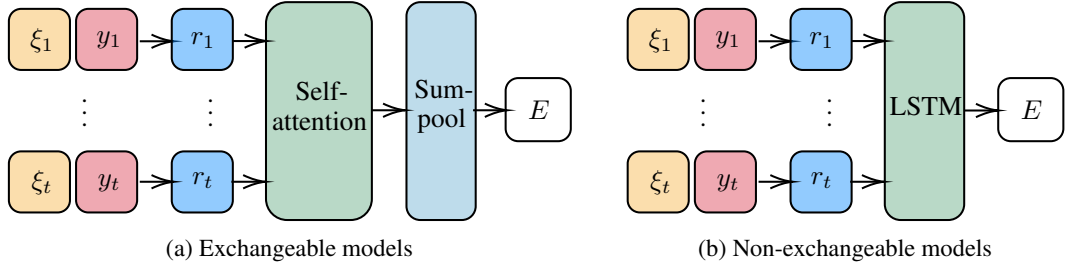


Figure 6: History encoder architectures for different classes of models. When conditional independence of the experiments holds, we use self-attention, followed by sum-pooling, making the history encoder permutation invariant. When experiments are not conditionally independent we use LSTM and only keep its last hidden state. We train two separate history encoders—one for the design network π_ϕ and one for the critic network U_ψ , although we note that all the weights except those in the head layers can be shared.

To the best of our knowledge, we are the first to propose a critic architecture that is tailored to BOED problems with exchangeable models. Previous work in the static BOED setting, where MI information objective is optimized with variational lower bounds and thus require the training of critics [e.g. 28, 63], did not discuss what an appropriate critic architecture might be. In particular, in all experiments [28, 63] use a generic architecture for both exchangeable and non-exchangeable problems. An expressive enough generic architecture should be able to obtain the optimal critic, and thus achieve a tight bound, however, the optimisation process will be considerably more difficult as the network needs to learn this key invariance structure. We therefore recommend using permutation invariant architectures whenever the model is exchangeable, especially if achieving tight bounds (and therefore learning an optimal critic) is of importance.

C.2 Further details on the history encoder

Figure 6 shows the history encoders we use in the policy network π_ϕ and the critic network U_ψ . First, we encode the individual design-outcome pairs, (ξ_t, y_t) , with an MLP, which gives us a vector of representations $r_t \in \mathbb{R}^m$, where m is the encoding dimension we have selected. The representations $\{r_i\}_{i=1}^t$ are row-stacked into a matrix R of dimension $t \times m$, which we then aggregate back to a vector of size m by an appropriate layer(s).

When conditional independence of the experiments holds, we apply 8-head self-attention, based on the Image Transformer [40] and as implemented by [14]. Applying self-attention leaves the dimension of the matrix R unchanged. We then apply sum-pooling across time t , which gives us the final encoding vector $E \in \mathbb{R}^m$.

When experiments are not conditionally independent, we pass the matrix R through an LSTM with two hidden layers and hidden state of size m (see the [LSTM module in Pytorch](#) for more details). The LSTM returns hidden state vectors associated with the history h_t for each t ; we keep the last hidden state of the last layer, which is our final encoding vector $E \in \mathbb{R}^m$.

In both cases the resulting encoding E is a vector of size m . It is passed through final fully connected "head" layers, which output either a design (in the case of the policy) or a vector (in the case of the critic). We train two separate history encoders—one for the design network π_ϕ and one for the critic network U_ψ , although we note that all the weights except those in the head layers can be shared.

D Experiments

D.1 Computational resources

All of the experiments were implemented in Python using open-source software. All estimators and models were implemented in PyTorch [41] (BSD license) and Pyro [6] (Apache License Version 2.0), whilst MIFlow [61] (Apache License Version 2.0) was used for experiment tracking and management. The self-attention architecture from [14] was used to implement the self-attention mechanisms in the

design and critic networks. For full details on package versions, environment set-up and commands for running the code, see instructions in the README.md file.

Experiments were ran on internal GPU clusters, consisting of GeForce RTX 3090 (24GB memory), GeForce RTX 2080 Ti (11GB memory) and GeForce GTX 1080 Ti GPUs (11GB memory).

The deployment-time of iDAD (Table 4) was estimated on a lightweight CPU machine with the following specifications

Processor	2.8 GHz Quad-Core Intel Core i7
Memory	16 GB
Operating system	macOS Big Sur v11.2.3

D.2 CO2 Emission Related to Experiments

Experiments were conducted using a private infrastructure, which has an estimated carbon efficiency of 0.432 kgCO₂eq/kWh. A cumulative of 160 hours of computation was performed on hardware of type RTX 2080 Ti (TDP of 250W), or similar. The training time of each experiment (including the baselines that require optimization), took on average between 1-3 GPU hours, depending on the number of experiments T .

Total emissions are estimated to be 17.28 kgCO₂eq of which 0% was directly offset.

Estimations were conducted using the [Machine Learning Impact calculator](#) presented in [31].

D.3 Traditional sequential BOED with variational posterior estimator

The variational posterior estimator from [15] is based on the Barbar-Agakov lower bound [4], which takes the form

$$\mathcal{L}^{\text{post}}(\xi, q_\psi) = \mathbb{E}_{p(\theta)p(y|\theta, \xi)} \left[\log \frac{q_\psi(\theta; y, \xi)}{p(\theta)} \right] \leq \mathcal{I}(\xi), \quad (102)$$

where q_ψ is any normalized distribution over the parameters θ . The bound is tight when $q_\psi(\theta; y, \xi) = p(\theta|y, \xi)$, i.e. if we can recover the true posterior. We assume mean-field variational family and optimize the parameters ψ by maximizing the bound (102) using stochastic gradient schemes. Simultaneously we optimize the bound with respect to the design variable ξ to select the optimal design ξ^* . At the inference stage, denoting by y^* the outcome of experiment ξ^* , we obtain an approximate posterior by evaluating $q_\psi(\theta; y^*, \xi^*)$, i.e. we reuse the learnt variational posterior. We repeat this process at each stage of the experiments by substituting the the approximate posterior, $q_\psi(\theta; y^*, \xi^*)$, as the prior in (102).

D.4 Location Finding

In this experiment we have K hidden objects (*sources*) in $\mathbb{R}^2$ and we wish to learn their locations, $\theta = \{\theta_1, \dots, \theta_K\}$. The number of sources, K , is assumed to be known. Each source emits a signal with intensity obeying the inverse-square law. Put differently, if a source is located at θ_k and we perform a measurement at a point ξ , the signal strength emitted from that source only will be proportional to $\frac{1}{\|\theta_k - \xi\|^2}$. The total intensity at location ξ , emitted from all K sources, is a superposition of the individual ones

$$\mu(\theta, \xi) = b + \sum_{k=1}^K \frac{\alpha_k}{m + \|\theta_k - \xi\|^2}, \quad (103)$$

where α_k can be known constants or random variables, $b > 0$ is a constant background signal and m is a constant, controlling the maximum signal.

We place a standard normal prior on each of the location parameters θ_k and we observe the log-total intensity with some Gaussian noise. We therefore have the following prior and likelihood:

$$\theta_k \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0_d, I_d) \quad \log y \mid \theta, \quad \xi \sim \mathcal{N}(\log \mu(\theta, \xi), \sigma^2) \quad (104)$$

D.4.1 Training details

All our experiments are performed with the following model hyperparameters

Parameter	Value
Number of sources, K	2
α_k	$1 \ \forall k$
Max signal, m	10^{-4}
Base signal, b	10^{-1}
Observation noise scale, σ	0.5

The architecture of the design network π_ϕ used in Table 2 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	3	3	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-
Attention	8 heads	64	64	-

The output of the encoder, $R(h_t)$, is fed into an emitter network, for which we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	64	64	-
H1	Fully connected	256	256	ReLU
H2	Fully connected	64	64	ReLU
Output	Fully connected	2	2	-

The architecture of the critic network U_ψ used in Table 2 and all its hyperparameters are in the tables that follow. First, the encoder network of the latent variables is:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	4	4	-
H1	Fully connected	16	16	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-

For the design-outcome pairs encoder we use the same architecture as in the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	3	3	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-
Attention	8 heads	64	64	-

The output of the encoder, $R(h_t)$, is fed into fully connected head layers:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	64	64	-
H1	Fully connected	1024	1024	ReLU
H2	Fully connected	512	512	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	64	64	-

The optimisation was performed with Adam [26] with ReduceLROnPlateau learning rate scheduler, with the following hyperparameters:

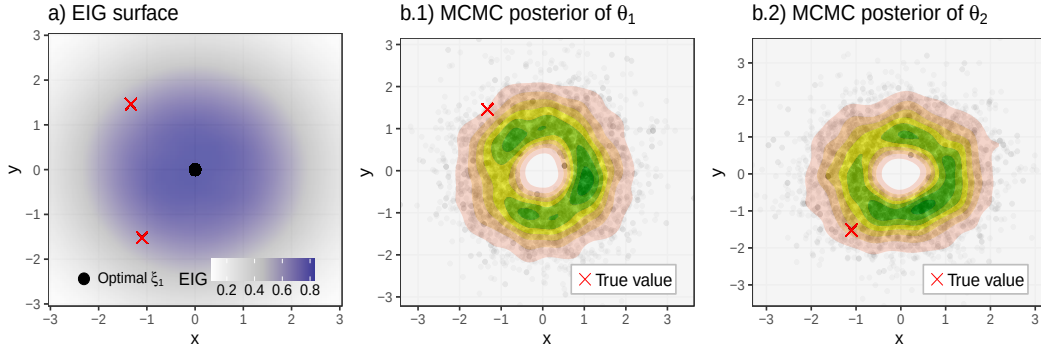


Figure 7: a): EIG surface induced by the prior; b) Samples from $p(\theta|\xi_1, y_1)$ —the posterior distribution of the locations, after performing experiment ξ_1 and observing y_1 , along with a KDE.

Parameter	iDAD, InfoNCE	iDAD, NWJ
Batch size	2048	2048
Number of contrastive/negative samples	2047	2047
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0005	0.0005
LR annealing factor	0.8	0.8
LR annealing frequency (if no improvement)	2000	2000

D.4.2 Performance of the variational baseline

As we saw in Table 2, this variational approach to (myopic) adaptive BOED performed very poorly, despite its large computational budget. The likely reason for that is that the mean-field variational approximation cannot adequately capture the complex non-Gaussian posterior of this problem. Figure 7 clearly demonstrates this: before any data is observed it is optimal to sample at the origin (since the prior is centered at it). After observing a low signal (the locations in this example are not close to the origin), we can only conclude that the sources are not within a small radius of the origin, but anywhere outside of it would be a plausible location, as indeed indicated by the fitted posteriors.

D.4.3 Hyperparameter selection

We did not perform extensive hyperparameters search; in particular, the network sizes were guided by two hyperparameters: hidden-dimension ($HD = 512$) and encoding dimension ($ED = 64$). We set-up all the networks to scale up with the number of experiments as follows:

- Design-outcome encoder has three hidden layers of sizes $[64, HD, ED]$.
- Design emitter network has three hidden layers of sizes $[HD/2, ED, 2]$, where 2 is the dimension of the design variable.
- The latent encoder for the critic network has four hidden layers of sizes $[16, 64, HD, ED]$.
- The critic design-outcome encoder’s head layer has four hidden layers of sizes $[HD \times \log(T), HD \times \log(T)/2, HD, ED]$.

Since our multi-head attention layer has 8 heads, the encoding dimension we use has to be a multiple of 8. In addition to $ED = 64$ we tried $ED = 32$ which provided marginally worse results. We did not try other values for these hyperparameters.

For the learning rate, we tried 0.001, which was too high, as well as 0.0005 (which we selected) and 0.0001 (which yielded very similar results).

We performed similar level of hyperparameter tuning for all trainable baselines as well (DAD, MINEBED and SG-BOED).

Table 6: Upper bounds on the total information, $\mathcal{I}_{10}(\pi)$, for the location finding experiment in Section 5.1. The bounds were estimated using $L = 5 \times 10^5$ contrastive samples. Errors indicate ± 1 s.e. estimated over 4096 histories (128 for variational). Lower bounds are presented in Table 2.

Method \ θ dim.	4D	6D	10D	20D
Random	4.794 ± 0.041	3.506 ± 0.004	1.895 ± 0.003	0.552 ± 0.001
MINEBED	5.522 ± 0.028	4.229 ± 0.029	2.459 ± 0.029	0.801 ± 0.019
SG-BOED	5.549 ± 0.028	4.220 ± 0.030	2.455 ± 0.029	0.803 ± 0.019
Variational	4.644 ± 0.146	3.626 ± 0.167	2.181 ± 0.152	0.669 ± 0.097
iDAD (NWJ)	7.806 ± 0.050	5.851 ± 0.041	3.264 ± 0.039	0.877 ± 0.022
iDAD (InfoNCE)	7.863 ± 0.043	6.068 ± 0.039	3.257 ± 0.040	0.872 ± 0.020
DAD	8.034 ± 0.038	6.310 ± 0.031	3.358 ± 0.040	0.953 ± 0.022

Table 7: Upper and lower bounds on the total information, $\mathcal{I}_{20}(\pi)$, for the location finding experiment in 2D from Section 5.1. The bounds were estimated using $L = 5 \times 10^5$ contrastive samples. Errors indicate ± 1 s.e. estimated over 4096 histories.

Method	Lower bound	Upper bound
Random	7.000 ± 0.034	7.020 ± 0.034
MINEBED	7.672 ± 0.030	7.690 ± 0.031
SG-BOED	7.701 ± 0.030	7.728 ± 0.031
iDAD (NWJ)	9.961 ± 0.033	10.372 ± 0.048
iDAD (InfoNCE)	10.075 ± 0.032	10.463 ± 0.043
DAD	10.424 ± 0.031	10.996 ± 0.049

D.4.4 Further ablation studies

Scalability with number of experiments. We first demonstrate that iDAD can scale to a larger number of experiments T . We train policy networks to perform $T = 20$ experiments and compare them to baselines in Table 7. We omit the variational baseline as it is too computationally costly to run for a large enough number of histories, and as we saw in the previous subsection, it is not particularly suited to this model.

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters (described in Subsection D.4.1) we used to produce the results of the location finding experiment in 2D (Table 2 in the main text). We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Estimator	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
InfoNCE	Lower	7.826	7.682	7.856	7.713	7.804	7.776	0.034
InfoNCE	Upper	7.933	7.791	7.856	7.807	7.925	7.862	0.029
NWJ	Lower	7.820	7.545	7.592	7.555	7.691	7.641	0.052
NWJ	Upper	7.976	7.640	7.669	7.651	7.800	7.747	0.064

We can see that the iDAD networks trained with InfoNCE are highly stable, with the 5 additional runs achieving very similar mutual information values to each other and to the iDAD network used the report the results in the main paper. The performance of the iDAD networks trained with the NWJ bound is more variable and empirically achieve slightly lower average value of mutual information. This higher variance is in-line with the discussion in Section B.

We similarly verify the robustness of the static baselines, reporting the results in the table below:

Table 8: Ablation study on the performance of iDAD as a function of training time for the location finding experiment.

Training budget	MI lower bound
0.1%	3.38
1.0%	6.09
2.0%	6.46
4.0%	6.81
8.0%	7.08
16.0%	7.33
32.0%	7.56
64.0%	7.78
100.0%	7.82

Estimator	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
SG-BOED	Lower	5.537	5.536	5.473	5.523	5.518	5.517	0.013
SG-BOED	Upper	5.553	5.548	5.491	5.541	5.531	5.533	0.012
MINEBED	Lower	5.460	5.506	5.553	5.539	5.565	5.524	0.021
MINEBED	Upper	5.473	5.526	5.567	5.554	5.574	5.540	0.022

Performance sensitivity to errors in the policy. Finally, we investigate the effect of slight errors in the design policy network. To this end, we look at the performance achieved by partially trained design networks (there will be some errors or inaccuracies in networks that were not trained until convergence). Table 8 shows the performance of iDAD as a function of training time, demonstrating that small errors in the network only lead to small drops in performance.

In detail, our results show that with just 8% of the total training budget, this slightly inaccurate network still performs relatively well, achieving total mutual information of 7.1, compared to the fully trained network that reached 7.8. We also highlight that iDAD outperforms all baselines with as little as 1% of the total training budget (the best performing baseline achieves mutual information of 5.5, see Table 2).

D.5 PK model

The drug concentration z , measured ξ hours after administering it, and the corresponding noisy observation y are given by

$$z(\xi; \theta) = \frac{D_V}{V} \frac{k_\alpha}{k_\alpha - k_e} [e^{-k_e \xi} - e^{-k_\alpha \xi}], \quad y(\xi; \theta) = z(\xi; \theta)(1 + \epsilon) + \eta \quad (105)$$

where $\theta = (k_\alpha, k_e, V)$, $D_V = 400$ is a constant, $\epsilon \sim \mathcal{N}(0, 0.01)$ is multiplicative noise to account for heteroscedasticity and $\eta \sim \mathcal{N}(0, 0.1)$ is an additive observation noise. Since both noise sources are Gaussian, the observation likelihood is also Gaussian i.e.

$$y(\xi; \theta) \sim \mathcal{N}(z(\xi; \theta), 0.01z(\xi; \theta)^2 + 0.1) \quad (106)$$

The prior for the parameters θ that we used

$$\log \theta \sim \mathcal{N} \left(\begin{bmatrix} \log 1 \\ \log 0.1 \\ \log 20 \end{bmatrix}, \begin{bmatrix} 0.05 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.05 \end{bmatrix} \right) \quad (107)$$

D.5.1 Training details

The architecture of the design network π_ϕ used for Figure 3 and 4 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-
Attention	8 heads	32	32	-

The outputs of the encoder, $\{R(h_t)\}_{t=1}^T$, are summed and the resulting vector (of dimension 32) is fed into an emitter network, for which we used the following:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_t)$	32	32	-
H1	Fully connected	256	256	ReLU
H2	Fully connected	32	32	ReLU
Output	Fully connected	1	1	Sigmoid

The architecture of the critic network U_ψ used in Figures 3 and 4 and all its hyperparameters are in the following tables. For the encoder of the design-outcome pairs we used the same architecture as for the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	64	64	ReLU
H2	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-
Attention	8 heads	32	32	-

The resulting pooled representation, $R(h_T)$ is fed into fully connected critic head layers with the following architecture:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$R(h_T)$	32	32	-
H1	Fully connected	512	512	ReLU
H2	Fully connected	256	256	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

Finally, for the latent variable encoder network we used:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	3	3	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The optimisation was performed with Adam [26] with the following hyperparameters:

Parameter	iDAD, InfoNCE	iDAD, NWJ
Batch size	1024	1024
Number of contrastive/negative samples	1023	1023
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0001	0.0001
LR annealing factor	0.8	0.5
LR annealing frequency (if no improvement)	2000	2000

Table 9: Upper and lower bounds on the total information, $\mathcal{I}_5(\pi)$, for the pharmacokinetic experiment. Errors indicate ± 1 s.e. estimated over 4096 (126 for variational) histories and $L = 5 \times 10^5$.

Method	Lower bound	Upper bound	Deployment time
Random	2.523 ± 0.033	2.523 ± 0.033	N/A
Equal interval	2.651 ± 0.022	2.651 ± 0.022	N/A
MINEBED	2.955 ± 0.030	2.956 ± 0.030	N/A
SG-BOED	2.985 ± 0.027	2.985 ± 0.027	N/A
Variational	2.683 ± 0.093	2.683 ± 0.093	505.4 $\pm 1\%$
IDAD (NWJ)	3.163 ± 0.023	3.163 ± 0.023	0.007 $\pm 7\%$
IDAD (InfoNCE)	3.200 ± 0.024	3.200 ± 0.024	0.007 $\pm 8\%$
DAD	3.234 ± 0.023	3.234 ± 0.023	0.002 $\pm 7\%$

Table 10: Upper and lower bounds on the total information, $\mathcal{I}_{10}(\pi)$, for the pharmacokinetic experiment. Errors indicate ± 1 s.e. estimated over 4096 (126 for variational) histories and $L = 5 \times 10^5$.

Method	Lower bound	Upper bound	Deployment time
Random	3.344 ± 0.034	3.345 ± 0.034	N/A
Equal interval	3.422 ± 0.026	3.423 ± 0.026	N/A
MINEBED	3.849 ± 0.034	3.849 ± 0.034	N/A
SG-BOED	3.824 ± 0.034	3.824 ± 0.034	N/A
Variational	3.624 ± 0.099	3.624 ± 0.099	1055.2 $\pm 8\%$
IDAD (NWJ)	4.034 ± 0.025	4.034 ± 0.025	0.007 $\pm 6\%$
IDAD (InfoNCE)	4.045 ± 0.026	4.045 ± 0.026	0.007 $\pm 5\%$
DAD	4.116 ± 0.024	4.117 ± 0.024	0.007 $\pm 8\%$

D.5.2 Hyperparameter selection

Hyperparameter selection was done in a way similar to the Location Finding experiment (see D.4.3). We tried encoding dimensions $ED = 32, 64$ and selected the smaller size as there were no clear benefits to larger networks (relatively speaking, this is an easier model than the location finding). We used the same hidden dimension, i.e. $HD = 512$. In terms of learning rates, we tried 0.0001, 0.0005 and 0.001; we found 0.0001 to be appropriate, although NWJ bound was exhibiting high variance, so used a smaller learning rate annealing factor for that network (0.5 vs 0.8 for InfoNCE). We performed similar level of hyperparameter tuning for all trainable baselines as well (DAD, MINEBED and SG-BOED).

D.5.3 Further results

Table 9 reports the results shown in Figure 3c), along with the corresponding upper bounds and deployment times, while Table 10 reports the results for $T = 10$.

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters we used to produce the results of the pharmacokinetic experiment (Figure 3c) and corresponding Table 9). We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
iDAD, InfoNCE	Lower	3.209	3.165	3.198	3.221	3.128	3.185	0.019
iDAD, InfoNCE	Upper	3.210	3.166	3.201	3.223	3.130	3.186	0.019
iDAD, NWJ	Lower	3.034	3.049	2.608	3.149	3.082	3.034	0.107
iDAD, NWJ	Upper	3.034	3.049	2.609	3.150	3.083	3.034	0.107

We repeat the same procedure for the static baselines. The results reported in the table below demonstrate the training stability of these baselines as well.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
SG-BOED	Lower	2.932	2.452	2.448	2.991	2.962	2.757	0.140
SG-BOED	Upper	2.932	2.453	2.449	2.992	2.962	2.757	0.140
MINEBED	Lower	2.912	2.213	3.014	2.092	2.941	2.634	0.221
MINEBED	Upper	2.914	2.213	3.015	2.092	2.942	2.635	0.222

D.6 SIR Model

Generally speaking, the SIR model advocates that, within a fixed population of size N , susceptible individuals $S(\tau)$, where τ is time, can become infected and move to an infected state $I(\tau)$. The infected individuals can then recover from the disease and move to the recovered state $R(\tau)$. The dynamics of these events are governed by the infection rate β and recovery rate γ , which define the particular disease in question. In the context of BOED, the aim is generally to estimate these two model parameters by observing state populations at particular measurement times τ , which are the experimental design variables. The SIR model has been studied extensively in the context of BOED, e.g. in [12, 27, 29, 30].

Stochastic versions of the SIR model are usually formulated via continuous-time Markov chains (CTMC), which can be simulated from via the Gillespie algorithm [2], yielding discrete state populations. However, iDAD requires us to differentiate through the sampling path of the state populations to the experimental designs, which is impossible if the simulated data is discrete as gradients are undefined. Thus, we here implement an alternative formulation of the stochastic SIR model that is based on stochastic differential equations (SDEs), as studied in [29], which yields continuous state populations that can be differentiated.

Following [29], let us first define a state population vector $\mathbf{X}(\tau) = (S(\tau), I(\tau))^\top$, where we can safely ignore the population of recovered $R(\tau)$ for modelling purposes because we assume that the total population stays fixed. The system of Itô SDEs that defines the stochastic SIR model is given by

$$d\mathbf{X}(\tau) = \mathbf{f}(\mathbf{X}(\tau))d\tau + \mathbf{G}(\mathbf{X}(\tau))d\mathbf{W}(\tau), \quad (108)$$

where $\mathbf{f}$ is a drift vector, $\mathbf{G}$ is a diffusion matrix and $\mathbf{W}(\tau)$ is a vector of independent Wiener processes. [29] showed that the drift vector and diffusion matrix are given by

$$\mathbf{f}(\mathbf{X}(\tau)) = \begin{pmatrix} -\beta \frac{S(\tau)I(\tau)}{N} \\ \beta \frac{S(\tau)I(\tau)}{N} - \gamma I(\tau) \end{pmatrix} \quad \text{and} \quad \mathbf{G}(\mathbf{X}(\tau)) = \begin{pmatrix} -\sqrt{\beta \frac{S(\tau)I(\tau)}{N}} & 0 \\ \sqrt{\beta \frac{S(\tau)I(\tau)}{N}} & -\sqrt{\gamma I(\tau)} \end{pmatrix}. \quad (109)$$

Given the system of Itô SDEs in (108), as well as the above drift vector and diffusion matrix, we can then simulate state populations $\mathbf{X}(\tau)$ by solving the SDE using finite-difference methods, such as e.g. the Euler-Maruyama method. See [29] for more information on the SDE-based SIR model, including derivations of the drift vector and diffusion matrix.

Importantly, we note that [29] further used the solutions of (108) as an input to a Poisson observation model, which increases the noise in simulated data. We here opt to simply use the solutions of (108) as data and do not consider an additional Poisson observational model.

D.6.1 Training details

As previously mentioned, the design variable for this model is the measurement time $\tau \in [0, 100]$. When solving the SDE with the Euler-Maruyama method, we discretize the time domain with a resolution of $\Delta\tau = 10^{-2}$. We here only use the number of infected $I(\tau)$ as the observed data, as others might be difficult to measure in reality. The total population is fixed at $N = 500$ and the initial conditions are $\mathbf{X}(\tau = 0) = (0, 2)^\top$. The model parameters β and γ have log-normal priors, i.e. $p(\beta) = \text{Lognorm}(0.50, 0.50^2)$ and $p(\gamma) = \text{Lognorm}(0.10, 0.50^2)$. Importantly, because solving SDEs is expensive, we pre-simulate our data on a time grid, store it in memory and then access the relevant data during training.

We present the network architectures and hyper-parameters corresponding to the $T = 5$ iDAD results shown in Table 5 of the main text. For the encoder of the design-outcome pairs we used:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The resulting representations, $\{R(h_t)\}_{t=1}^{T-1}$, are stacked into a matrix (as new design–outcome pairs are obtained) and fed into an emitter network, which contains an LSTM cell with two hidden layers. We only keep the last hidden state of the LSTM’s output and pass it through a final FC layer:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$\{R(h_t)\}_{t=1}^{T-1}$	$32 \times t$	$32 \times t$	-
H1 & H2	LSTM	32	32	-
H3	Fully connected	16	16	ReLU
Output	Fully connected	1	1	-

The architecture of the critic network U_ψ used in Table 5 and all its hyper-parameters are in the tables that follow. First, the encoder network of the latent variables is:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	θ	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

For the design–outcome pairs encoder we use the same architecture as in the design network, namely:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	ξ, y	2	2	-
H1	Fully connected	8	8	ReLU
H2	Fully connected	64	64	ReLU
H3	Fully connected	512	512	ReLU
Output	Fully connected	32	32	-

The outputs of the encoder, $\{R(h_t)\}_t$, are stacked and fed into an LSTM cell with two hidden layers. We only keep the last hidden state of the LSTM’s output and pass it through a FC layer:

Layer	Description	iDAD, InfoNCE	iDAD, NWJ	Activation
Input	$\{R(h_t)\}_{t=1}^{T-1}$	$32 \times t$	$32 \times t$	-
H1 & H2	LSTM	32	32	-
H3	Fully connected	16	16	ReLU
Output	Fully connected	32	32	-

The optimization was performed with Adam [26] with learning rate annealing with the following hyper-parameters:

Parameter	iDAD InfoNCE	iDAD, NWJ
Batch size	512	512
Number of contrastive/negative samples	511	511
Number of gradient steps	100000	100000
Initial learning rate (LR)	0.0005	0.0005
LR annealing factor	0.96	0.96
LR annealing frequency	1000	1000

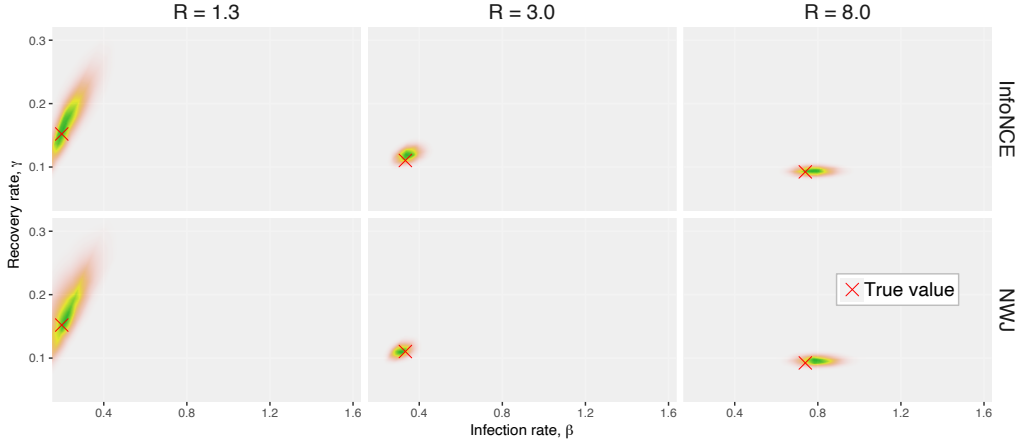


Figure 8: Approximate posteriors for the SIR model.

D.6.2 Further results

Different number of experiments T . In Table 11 we show lower bound estimates when applying iDAD with the InfoNCE lower bound to the SDE-based SIR model for different number of measurements T . The design network and critic architectures are the same as for $T = 5$. Table 11 shows that more measurements yield higher expected information gains, as one might intuitively expect. Furthermore, the increase in expected information gain saturates with increasing T , which is why we presented the results for $T = 5$ in the main text. The biggest increase, however, occurs from $T = 1$ to $T = 2$. This is intuitive, because the SIR model has two model parameters that we wish to estimate but we only gather one data point with one measurement. Hence, in order to accurately estimate both of these parameters, we would need at least 2 measurements, which is reflected in Table 11. We note that all of these numbers, with the exception of $T = 1$, are larger than those found by [29]. This increase in expected information gain may be explained by the fact that [29] use an additional Poisson observation model, which means that the resulting data are inherently noisier and less informative.

Table 11: InfoNCE lower bound estimates ($\pm$ s.e.) when applying iDAD to the SDE-based SIR model for different number of measurements T .

T	iDAD, InfoNCE	iDAD, NWJ
1	1.396 ± 0.018	1.417 ± 0.001
2	2.714 ± 0.019	2.699 ± 0.001
3	3.554 ± 0.021	3.515 ± 0.001
4	3.600 ± 0.018	3.749 ± 0.001
5	3.915 ± 0.020	3.869 ± 0.001
7	4.027 ± 0.019	3.911 ± 0.001
10	4.100 ± 0.020	4.019 ± 0.001

Training stability. To assess the robustness of the results and the stability of the training process, we trained 5 additional iDAD networks with each of the two bounds, using different seeds but the same hyperparameters we used to produce the results of Table 5 in the main text. We report upper and lower bounds on the mutual information along with their mean and standard error in the table below.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
iDAD, InfoNCE	Lower	3.900	3.919	3.919	3.901	3.887	3.906	0.007
iDAD, NWJ	Lower	3.872	3.838	3.854	3.883	3.848	3.859	0.009

We repeat the same procedure for the static baselines. The results reported in the table below demonstrate the training stability of these baselines as well.

Method	Bound	Run 1	Run 2	Run 3	Run 4	Run 5	Mean	SE
SG-BOED	Lower	3.713	3.765	3.767	3.764	3.739	3.749	0.012
MINEBED	Lower	3.373	3.438	3.376	3.379	3.420	3.397	0.015

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods
Publication Status	Published
Publication Details	Ivanova, D. R., Foster, A., Kleinegesse, S., Gutmann, M. U., & Rainforth, T. (2021). Implicit deep adaptive design: Policy-based experimental design without likelihoods. <i>Advances in Neural Information Processing Systems</i> , 34, 25785-25798.

Student Confirmation

Student Name:	Desi R Ivanova		
Contribution to the Paper	First author. - Led the development of the methodology and the writing of the paper. - Performed experiments in Sec 5.1 and 5.2, and contributed to the experiment in Sec 5.3. - Developed the codebase that was used for the empirical evaluations in the paper.		
Signature 	Date	18 April 2024	

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Dr Tom Rainforth		
Supervisor comments The contribution statement above is correct; Desi was the primary contributor to the project.		
Signature 	Date	19 April 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 4

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

This paper is under submission as as:

Desi R Ivanova*, Marcel Hedman*, Cong Guan and Tom Rainforth. Step-DAD:
Semi-Amortized Policy-Based Bayesian Experimental Design.

*Equal contribution

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

Desi R Ivanova^{* 1} Marcel Hedman^{* 1} Cong Guan² Tom Rainforth¹

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. Empirically, Step-DAD consistently demonstrates superior decision-making and robustness compared with current state-of-the-art BED methods, substantially improving the quality of the data gathered.

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 & Verdinelli, 1995; Ryan et al., 2016; Rainforth et al., 2024). 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 out-

comes 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, illustrated in Fig 1a, involves iterating between making design decisions by optimizing the EIG of the next experiment step, and updating the underlying model through Bayesian updates that condition on the data obtained so far. Unfortunately, this approach leads to sub-optimal design decisions, as it is a greedy, myopic, strategy that fails to plan ahead and consider future experiment steps (Huan & Marzouk, 2016; Foster, 2021). Furthermore, it requires substantial computation to be undertaken at each experiment iteration (in the form of the posterior updates and EIG optimizations), making it impractical for real-time applications (Rainforth et al., 2024).

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* that maps from experimental histories to new design decisions. This policy network is trained upfront before the experiment, then deployed to make design decisions automatically at run time. This provides a *fully amortized* approach that eliminates the need for any significant computation during the experiment itself, thereby enabling real-time, adaptive, and non-myopic design strategies that represent the current state-of-the-art in adaptive BED.

In principle, these fully amortized approaches can learn theoretically 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 can mean that data observed in practice can be highly distinct from the simulated data used to train the policy.

To address these limitations, and allow utilisation of any computation that is available during the experiment, we in-

^{*}Equal contribution ¹Department of Statistics, University of Oxford, UK ²Work undertaken whilst at the University of Oxford. Correspondence to: <desi.ivanova, marcel.hedman>@stats.ox.ac.uk.

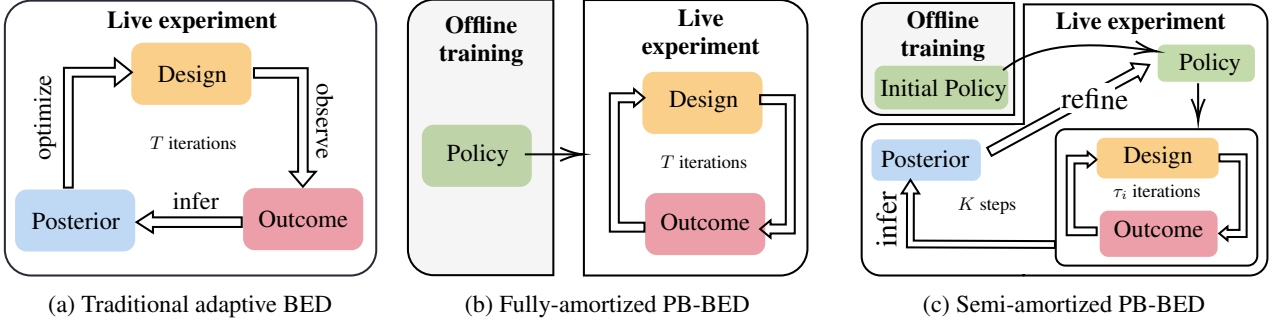


Figure 1. Overview of adaptive BED approaches. The traditional BED approach fits a posterior after each experiment iteration and optimizes for the next step best designs (i.e. greedily). Fully amortized policy-based BED approaches like DAD train a policy once offline, before the live experiment, then 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.

roduce 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 the traditional BED approach.

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 $p(\theta)p(y|\theta, \xi)$, describing the relationship between experimental outcomes y , controllable designs ξ and unknown parameters of interest θ , the goal is to select the experiment ξ that maximizes the expected information gain (EIG) about θ . The EIG, which is equivalent to the mutual information between parameters and outcomes, is the expected reduction in Shannon entropy from the prior to the posterior distribution of θ :

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

where $p(y|\xi) = \mathbb{E}_{p(\theta)}[p(y|\theta, \xi)]$ is the prior predictive distribution of our model.

2.1. Traditional Adaptive BED

BED becomes particularly powerful in adaptive experimental contexts, where we allow the future design decision at time t , ξ_t , to be informed by the data acquired up to that point, $h_{t-1} := (\xi_1, y_1), \dots, (\xi_{t-1}, y_{t-1})$, 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 maximization of the one-step ahead or *incremental* EIG for the next design

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

where $p(y|\xi_t, h_{t-1}) = \mathbb{E}_{p(\theta|h_{t-1})}[p(y|\theta, \xi_t, h_{t-1})]$ represents the marginal outcome distribution under our updated model. We use the superscript h_{t-1} to emphasize conditioning on the history currently available, setting $h_0 = \emptyset$. This is a *closed-loop* approach (Foster, 2022; Huan & Marzouk, 2016), explicitly integrating all of the acquired data to refine beliefs about θ and inform subsequent design decisions.

Whilst this traditional framework offers a principled and systematic 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 both of these steps must be conducted at each step of the experiment, 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 subsequent design choices, $\pi: h_{t-1} \mapsto \xi_t$. The optimal policy is now the one that maximizes the *total* EIG across the entire sequence of

T experiments (Foster et al., 2021; Shen & Huan, 2021)

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

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

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 specific policy that maximizes the incremental one-step-ahead EIG (2) at each iteration, $\pi(h_{t-1}) = \arg \max_{\xi_t} I_{t-1 \rightarrow t}^{h_{t-1}}(\xi_t)$.

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 & Gutmann, 2020) coupled with stochastic gradient ascent (SGA) schemes to directly train a policy network taking the form of a deep 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 $p(\theta)p(h_T|\theta,\pi)$. 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 step of the experiment, thereby allowing design decisions to be made almost instantly at deployment.

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 comes 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 design decisions, particularly in scenarios where the real-world experimental data significantly deviates from the simulated histories used during training of the policy. Two primary 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 & McGree, 2022; Go & Isaac, 2022). Moreover, even if the model is well-specified from a Bayesian inference perspective (Uppal & 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 3.1 (Decomposition of total EIG). *For any design policy π , the total EIG of a T -step experiment can be decomposed as*

$$\mathcal{I}_{1 \rightarrow T}(\pi) = \mathcal{I}_{1 \rightarrow \tau}(\pi) + \mathbb{E}_{p(h_\tau | \pi)}[\mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi)], \quad (5)$$

for any intermediate step $1 \leq \tau \leq T$, where

$$\mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi) = \mathbb{E}_{p(\theta | h_\tau)p(h_{\tau+1:T} | h_\tau, \theta, \pi)} \left[\log \frac{p(h_{\tau+1:T} | h_\tau, \theta, \pi)}{p(h_{\tau+1:T} | h_\tau, \pi)} \right]. \quad (6)$$

Proof. We can write the likelihood and marginal as

$$\begin{aligned} p(h_T | \theta, \pi) &= p(h_\tau | \theta, \pi)p(h_{\tau+1:T} | h_\tau, \theta, \pi) \\ p(h_T | \pi) &= p(h_\tau | \pi)p(h_{\tau+1:T} | h_\tau, \pi) \end{aligned}$$

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

$$\begin{aligned} \mathcal{I}_{1 \rightarrow T}(\pi) &= \mathbb{E}_{p(\theta)p(h_\tau | \theta, \pi)} \left[\log \frac{p(h_\tau | \theta, \pi)}{p(h_\tau | \pi)} \right] + \\ &\quad \mathbb{E}_{p(h_\tau | \pi)p(\theta | h_\tau)p(h_{\tau+1:T} | h_\tau, \theta, \pi)} \left[\log \frac{p(h_{\tau+1:T} | h_\tau, \theta, \pi)}{p(h_{\tau+1:T} | h_\tau, \pi)} \right] \\ &= \mathcal{I}_{1 \rightarrow \tau}(\pi) + \mathbb{E}_{p(h_\tau | \pi)}[\mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi)] \text{ as required.} \quad \square \end{aligned}$$

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 π_{h_τ} , which is trained to maximize the total *remaining* EIG, $\mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi)$, as defined in (6).

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 $p(\theta | h_\tau)$ 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 (6). 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.

It is important to 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 the available time is limited. In particular, as we will 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. As such, semi-amortized PB-BED still maintains large computational benefits over traditional adaptive BED.

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*, $\mathcal{T} = \tau_0, \tau_1, \dots, \tau_K$ —an increasing sequence defining the points at which the policy is refined. We adopt the convention $\tau_0 = 0$ and $h_0 = \emptyset$, 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 (6) to sequentially train the step policy $\pi^{s(\mathcal{T})}$ in a scalable manner. An overview of Step-DAD is presented in Algorithm 1.

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.

Algorithm 1 Overview of Step-DAD

Input: Generative model $p(\theta)p(y|\theta, \xi)$, experimental budget T , refinement schedule $\mathcal{T}=\{\tau_0, \tau_1, \dots, \tau_{K+1}\}$, 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 $h_0 = \emptyset$.
- 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** $k = 1, \dots, K + 1$ **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 $p(\theta | h_{\tau_k})$
 - ▷ Fine-tune policy $\pi_{h_{\tau_k}}$ by optimizing (7)
 - end**
- end**

4.1. Policy refinement

Due to its doubly intractable nature, the task of optimizing the remaining EIG, $\mathcal{I}_{\tau_k+1 \rightarrow T}^{h_{\tau_k}}(\pi)$, 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 & Gutmann, 2020) bounds, align with this requirement. For generative models with explicit likelihoods (implicit models are discussed in Appendix A), we, therefore, choose to employ the sPCE lower bound, defined as

$$\mathcal{L}_{\tau_k+1 \rightarrow T}^{\text{sPCE}}(\pi) = \mathbb{E} \left[\log \frac{p(h_{\tau_k+1:T} | h_{\tau_k} \theta_0, \pi)}{\frac{1}{L+1} \sum_{\ell=0}^L p(h_{\tau_k+1:T} | \theta_\ell, \pi)} \right]. \quad (7)$$

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 $\mathcal{L}_{\tau_k+1 \rightarrow T}^{\text{sPCE}}(\pi)$, with respect to the network parameters using standard stochastic gradient ascent (SGA) schemes (Robbins & Monro, 1951; Kingma & 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 (7) 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 example, only adjusting the last few layers.

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 & 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, as explored by Blau et al. (2022) and Lim et al. (2022). None of these 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 & Rainforth, 2017), variational bounds (Foster et al., 2019; 2020; Kleinegesse & Gutmann, 2020) and ratio estimation (Kleinegesse & 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 & McGree, 2020). Thanks to advancements in gradient-based methods, EIG estimation and optimization can be performed jointly in a SGA scheme (Huan & Marzouk, 2014; Foster et al., 2020; Kleinegesse & Gutmann, 2021).

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 & 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 & 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 & 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 the step policy $\pi^{s(T)}$ over π_{h_0} —the fully amor-

Table 1. Source Location Finding. Upper and lower bound estimates of total EIG. For Step-DAD we report best finetuning step, $\tau = 6$ with the rest in Table 8 in the Appendix. Errors indicate ± 1 s.e., computed over 16 (2048) histories for step methods (rest).

Method	Lower bound ($\uparrow$)	Upper bound ($\uparrow$)
Random	3.612 ± 0.012	3.613 ± 0.012
Static	3.945 ± 0.026	3.946 ± 0.026
Step-Static ($\tau = 6$)	3.974 ± 0.008	3.975 ± 0.008
DAD	6.771 ± 0.012	6.803 ± 0.013
Step-DAD ($\tau = 6$)	7.605 ± 0.078	7.609 ± 0.078

tized one. 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. Wherever appropriate, we include a **Random** design strategy serving as a naive, non-optimized benchmark, as well as **problem-specific** baselines if available.

Our main metric for assessing the quality of various design strategies is the **total EIG**, $\mathcal{I}_{1 \rightarrow T}(\pi)$, as given in (3). We approximate it via the sPCE lower bound (7) along with its upper bound counterpart—the sequential Nested Monte Carlo estimator (sNMC, Foster et al., 2021). For the remaining EIG (6), we leverage the decomposition from Proposition 3.1, estimating the expectation over the partial history h_τ with $N = 16$ realizations. Details on experimental setup, training and evaluation, are provided in Appendix B.

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 & 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. The full model is described in Appendix B.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 $\tau = 1, \dots, 9$, using importance sampling to approximate the posterior $p(\theta | h_\tau)$. Results presented in Table 1 highlight the best performing finetuning step, $\tau = 6$, at which Step-DAD surpasses all baseline methods. Table 8 in the Appendix details performances across all τ values, revealing that steps $\tau = 4, 5, 7$ exhibit performance statistically equivalent to that of $\tau = 6$.

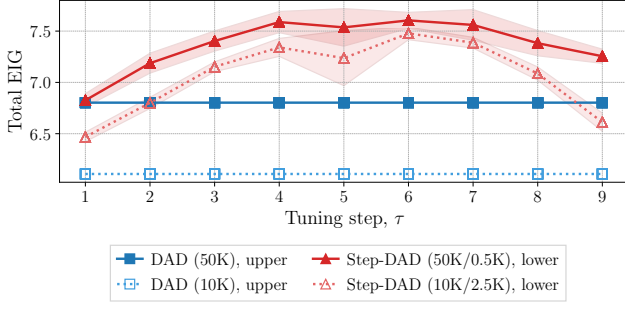


Figure 2. **Sensitivity to training budget** for location finding experiment. DAD policies are trained for 50K or 10K steps, Step-DAD policies are refined for 0.5K and 2.5K steps, resp. Step-DAD outperforms its respective DAD baseline for all τ . Errors show ± 1 s.e.

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 ($5\times$ 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 (dashed red line with triangle markers), consistently outperforms the DAD model trained on a 50K step budget (solid blue line with square markers), 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. The consistent positive EIG difference for Step-DAD, reported in Table 2, highlights the scalability of our method to higher-dimensional parameter spaces.

Table 2. **Location finding multiple sources.** Errors show ± 1 s.e.

θ dim	EIG difference	DAD, total EIG (upper)
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

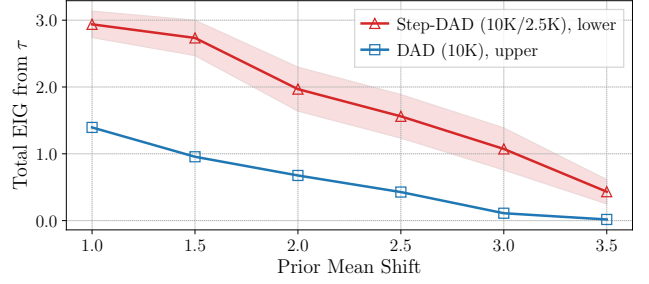


Figure 3. **Sensitivity to prior perturbations** in the source location finding experiment with one source and $T = 10$. Total EIG from $\tau = 6$ for Step-DAD remains robust despite increasing shift, whilst that for the DAD baseline drops to zero.

6.2. Robustness to prior perturbations

In BED, selecting an appropriate prior is critical, particularly in scenarios where data is sparse and therefore the prior has significant influence on the posterior (Simchowitz et al., 2021; Go & Isaac, 2022). We evaluate the robustness of Step-DAD when the prior distribution at test time, $\tilde{p}(\theta)$, diverges from the one utilized during the offline training phase, $p(\theta)$. This scenario mirrors real-world situations where the conditions during the live experiment may not match those assumed during the training of the fully amortized policy.

We assess robustness using the total EIG **under an alternative model** defined as

$$\mathcal{I}_{\tilde{p}(\theta)}(\pi) := \mathbb{E}_{\tilde{p}(\theta)p(h_T|\theta,\pi)} \left[\log \frac{p(h_T|\theta,\pi)}{p(h_T|\pi)} \right], \quad (8)$$

where the $p(h_T|\pi) = \mathbb{E}_{\tilde{p}(\theta)}[p(h_T|\theta,\pi)]$ and $\tilde{p}(\theta)$ is the perturbed prior distribution.

The results for the source location finding design problem are shown in Figure 3. It reveals that Step-DAD consistently outperforms the DAD baseline across all degrees of prior shift we consider, with the EIG for DAD decreasing to essentially zero with the increased prior shift, whilst Step-DAD is 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.

6.3. 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 economics and has been widely applied to study important social and individual behaviors (Critchfield & 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 be-

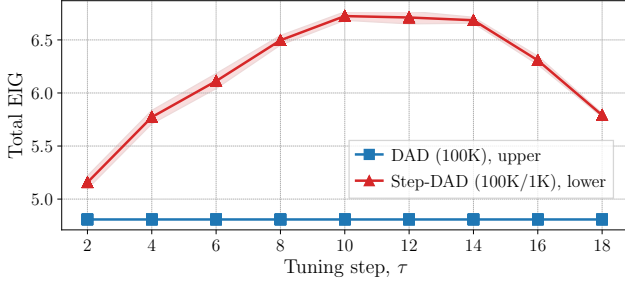


Figure 4. **Hyperbolic temporal discounting.** EIG improvement of Step-DAD over DAD after fine-tuning the policy at step τ . The fully amortized DAD network is trained for 100K steps, whilst subsequent policy refinement is performed for 1K steps.

Table 3. **Hyperbolic temporal discounting.** Upper and lower bound estimates of total EIG, $\mathcal{I}_{1 \rightarrow 20}(\pi)$. Errors indicate ± 1 s.e., computed over 16 (2048) histories for the step methods (rest). All baselines except DAD are as reported in Foster et al. (2021).

Method	Upper bound ($\uparrow$)	Lower bound ($\uparrow$)
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 ($\tau=10$)	6.711 ± 0.040	6.721 ± 0.040

haviour (Holt et al., 2003; Story et al., 2014).

An individual’s time delay preference is typically measured by asking them 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 outcome of the experiment y is the participant’s decision to either *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 original 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

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 $T = 40$. Error bars indicate ± 1 s.e. computed over 16 histories.

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

derived from psychology research (Kirby, 2009; Frye et al., 2016; Vincent & Rainforth, 2017) and traditional BED approaches such as BADapted (Vincent & 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. This highlights the robustness and flexibility of our method in extending experimental horizons.

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 policies. 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.

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.

References

- Atkinson, A., Donev, A., and Tobias, R. *Optimum Experimental Designs, with SAS*. Oxford University Press, 2007. 1
- Betancourt, M. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017. 6
- Bickel, W. K., Freitas-Lemos, R., Tomlinson, D. C., Craft, W. H., Keith, D. R., Athamneh, L. N., Basso, J. C., and Epstein, L. H. Temporal discounting as a candidate behavioral marker of obesity. *Neuroscience & Biobehavioral Reviews*, 129:307–329, 2021. 7
- Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 2018. 12
- Blau, T., Bonilla, E., Dezfouli, A., and Chadès, I. Optimizing sequential experimental design with deep reinforcement learning. *arXiv preprint arXiv:2202.00821*, 2022. 1, 3, 5
- Chaloner, K. and Verdinelli, I. Bayesian experimental design: A review. *Statistical Science*, pp. 273–304, 1995. 1
- Critchfield, T. S. and Kollins, S. H. Temporal discounting: Basic research and the analysis of socially important behavior. *Journal of applied behavior analysis*, 34(1): 101–122, 2001. 7
- Del Moral, P., Doucet, A., and Jasra, A. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006. 6
- Drovandi, C. C., McGree, J. M., and Pettitt, A. N. A sequential monte carlo algorithm to incorporate model uncertainty in bayesian sequential design. *Journal of Computational and Graphical Statistics*, 23(1):3–24, 2014. 6
- Farquhar, S., Gal, Y., and Rainforth, T. On statistical bias in active learning: How and when to fix it. *arXiv preprint arXiv:2101.11665*, 2021. 6
- Foster, A. *Variational, Monte Carlo and Policy-Based Approaches to Bayesian Experimental Design*. PhD thesis, University of Oxford, 2022. 2
- Foster, A., Jankowiak, M., Bingham, E., Horsfall, P., Teh, Y. W., Rainforth, T., and Goodman, N. Variational Bayesian Optimal Experimental Design. In *Advances in Neural Information Processing Systems* 32, pp. 14036–14047. Curran Associates, Inc., 2019. 2, 3, 5, 6
- Foster, A., Jankowiak, M., O’Meara, M., Teh, Y. W., and Rainforth, T. A unified stochastic gradient approach to designing bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pp. 2959–2969. PMLR, 2020. 3, 6
- Foster, A., Ivanova, D. R., Malik, I., and Rainforth, T. Deep adaptive design: Amortizing sequential bayesian experimental design. *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR 139, 2021. 1, 2, 3, 5, 6, 8, 12, 14
- Foster, A. E. *Variational, Monte Carlo and Policy-Based Approaches to Bayesian Experimental Design*. PhD thesis, University of Oxford, 2021. 1
- Frye, C. C., Galizio, A., Friedel, J. E., DeHart, W. B., and Odum, A. L. Measuring delay discounting in humans using an adjusting amount task. *JoVE (Journal of Visualized Experiments)*, (107):e53584, 2016. 8
- Go, J. and Isaac, T. Robust expected information gain for optimal bayesian experimental design using ambiguity sets. In *Uncertainty in Artificial Intelligence*, pp. 728–737. PMLR, 2022. 3, 6, 7
- Holt, D. D., Green, L., and Myerson, J. Is discounting impulsive?: Evidence from temporal and probability discounting in gambling and non-gambling college students. *Behavioural processes*, 64(3):355–367, 2003. 8
- Huan, X. and Marzouk, Y. Gradient-based stochastic optimization methods in bayesian experimental design. *International Journal for Uncertainty Quantification*, 4(6), 2014. 6
- Huan, X. and Marzouk, Y. M. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013. 6
- Huan, X. and Marzouk, Y. M. Sequential bayesian optimal experimental design via approximate dynamic programming. *arXiv preprint arXiv:1604.08320*, 2016. 1, 2, 5
- Ivanova, D. R., Foster, A., Kleinegesse, S., Gutmann, M., and Rainforth, T. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. In *Advances in Neural Information Processing Systems*, volume 34, pp. 25785–25798. Curran Associates, Inc., 2021. URL <https://proceedings.neurips.cc/paper/2021/file/d811406316b669ad3d370d78b51b1d2e-Paper.pdf>. 1, 3, 5
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

- Kirby, K. N. One-year temporal stability of delay-discount rates. *Psychonomic bulletin & review*, 16(3):457–462, 2009. 8
- Kleinegesse, S. and Gutmann, M. Efficient Bayesian experimental design for implicit models. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pp. 1584–1592. PMLR, 2019. 6
- Kleinegesse, S. and Gutmann, M. 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, pp. 5316–5326. PMLR, 2020. 3, 5, 6
- Kleinegesse, S. and Gutmann, M. U. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds. *arXiv preprint arXiv:2105.04379*, 2021. 6
- Kleinegesse, S., Drovandi, C., and Gutmann, M. U. Sequential Bayesian Experimental Design for Implicit Models via Mutual Information. *Bayesian Analysis*, pp. 1 – 30, 2021. doi: 10.1214/20-BA1225. 6
- Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., and Teh, Y. W. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pp. 3744–3753. PMLR, 2019. 5
- Lim, V., Novoseller, E., Ichnowski, J., Huang, H., and Goldberg, K. Policy-based bayesian experimental design for non-differentiable implicit models. *arXiv preprint arXiv:2203.04272*, 2022. 1, 3, 5
- Lindley, D. V. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pp. 986–1005, 1956. 1, 2
- Lindley, D. V. *Bayesian statistics, a review*, volume 2. SIAM, 1972. 1
- Lintusaari, J., Gutmann, M., Dutta, R., Kaski, S., and Corander, J. Fundamentals and recent developments in approximate Bayesian computation. *Systematic Biology*, 66(1): e66–e82, January 2017. 6
- MacKay, D. J. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992. 1
- Mazur, J. E. An adjusting procedure for studying delayed reinforcement. *Commons, ML.; Mazur, JE.; Nevin, JA*, pp. 55–73, 1987. 8, 14
- McClelland, J., Dalton, B., Kekic, M., Bartholdy, S., Campbell, I. C., and Schmidt, U. A systematic review of temporal discounting in eating disorders and obesity: Behavioural and neuroimaging findings. *Neuroscience & Biobehavioral Reviews*, 71:506–528, 2016. 7
- Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020. 5
- Myung, J. I., Cavagnaro, D. R., and Pitt, M. A. A tutorial on adaptive design optimization. *Journal of mathematical psychology*, 57(3-4):53–67, 2013. 1, 6
- Nguyen, X., Wainwright, M. J., and Jordan, M. I. 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. 5, 12
- Overstall, A. and McGree, J. 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. 6
- Overstall, A. and McGree, J. Bayesian decision-theoretic design of experiments under an alternative model. *Bayesian Analysis*, 17(4):1021–1041, 2022. 3, 6
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. 12
- Price, D. J., Bean, N. G., Ross, J. V., and Tuke, J. An induced natural selection heuristic for finding optimal bayesian experimental designs. *Computational Statistics & Data Analysis*, 126:112–124, 2018. 6
- Rainforth, T. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, University of Oxford, 2017. 6
- Rainforth, T., Cornish, R., Yang, H., Warrington, A., and Wood, F. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pp. 4267–4276. PMLR, 2018. 2, 5
- Rainforth, T., Foster, A., Ivanova, D. R., and Bickford Smith, F. Modern bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024. 1

- Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pp. 1278–1286, 2014. 5
- Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951. 5
- Ryan, E. G., Drovandi, C. C., McGree, J. M., and Pettitt, A. N. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016. 1, 2
- Shen, W. and Huan, X. 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
- Sheng, X. and Hu, Y. H. 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. 6
- Simchowitz, M., Tosh, C., Krishnamurthy, A., Hsu, D. J., Lykouris, T., Dudik, M., and Schapire, R. E. Bayesian decision-making under misspecified priors with applications to meta-learning. *Advances in Neural Information Processing Systems*, 34:26382–26394, 2021. 7
- Sisson, S., Fan, Y., and Beaumont, M. *Handbook of Approximate Bayesian Computation*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, 2018. ISBN 9781351643467. 6
- Story, G. W., Vlaev, I., Seymour, B., Darzi, A., and Dolan, R. J. Does temporal discounting explain unhealthy behavior? a systematic review and reinforcement learning perspective. *Frontiers in behavioral neuroscience*, 8:76, 2014. 8
- Tate, L. M., Tsai, P.-F., Landes, R. D., Rettiganti, M., and Lefler, L. L. Temporal discounting rates and their relation to exercise behavior in older adults. *Physiology & behavior*, 152:295–299, 2015. 7
- Thomas, O., Dutta, R., Corander, J., Kaski, S., and Gutmann, M. U. Likelihood-free inference by ratio estimation. *arXiv preprint arXiv:1611.10242*, 2016. 6
- Uppal, R. and Wang, T. Model misspecification and underdiversification. *The Journal of Finance*, 58(6):2465–2486, 2003. 3
- van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5, 12
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017. 5
- Vincent, B. T. Hierarchical bayesian estimation and hypothesis testing for delay discounting tasks. *Behavior research methods*, 48(4):1608–1620, 2016. 8, 14
- Vincent, B. T. and Rainforth, T. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *PsyArXiv preprint*, 2017. 6, 8
- Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 5
- Zaharia, M., Chen, A., Davidson, A., Ghodsi, A., Hong, S., Konwinski, A., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., Xie, F., and Zumar, C. Accelerating the machine learning lifecycle with MLflow. *IEEE Data Eng. Bull.*, 41:39–45, 2018. 12
- Zaheer, M., Kottur, S., Ravanbakhsh, S., Póczos, B., Salakhutdinov, R., and Smola, A. J. Deep sets. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, 2017. 5
- Zhang, C., Bütepage, J., Kjellstrom, H., and Mandt, S. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):2008–2026, 2018. 6

A. Further EIG bounds

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

$$\mathcal{U}_{\tau_k+1 \rightarrow T}^{\text{sNMC}}(\pi) := \mathbb{E} \left[\log \frac{p(h_{\tau_k+1:T} | h_{\tau_k} \theta_0, \pi)}{\frac{1}{L} \sum_{\ell=1}^L p(h_{\tau_k+1:T} | \theta_\ell, \pi)} \right], \quad (9)$$

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

$$\mathcal{L}_{\text{InfoNCE}}(\pi, U; L) := \mathbb{E}_{p(\theta_0)p(h_T|\theta_0, \pi)} \mathbb{E}_{p(\theta_1:L)} \left[\log \frac{\exp(U(h_T, \theta_0))}{\frac{1}{L+1} \sum_{i=0}^L \exp(U(h_T, \theta_i))} \right], \quad (10)$$

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

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

$$\mathcal{L}_{\text{NWJ}}(\pi, U) := \mathbb{E}_{p(\theta)p(h_T|\theta, \pi)} [U(h_T, \theta) - e^{-1} \mathbb{E}_{p(\theta)p(h_T|\pi)} [\exp(U(h_T, \theta))]] \quad (11)$$

B. Experiment details

B.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, MIFlow (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 ran on a single GPU.

B.2. Evaluation details

When evaluating fully amortized policies, we employ the sPCE (7) lower bound and sNMC (9) 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 $p(\theta)p(h_T | \theta, \pi)$. To approximate this quantity for Step-DAD efficiently, we use

$$\Delta \mathcal{I}(\pi^{s(\tau)}, \pi_{h_0}) := \mathcal{I}_{1 \rightarrow T}(\pi^{s(\tau)}) - \mathcal{I}_{1 \rightarrow T}(\pi_{h_0}) \quad (12)$$

$$= \mathbb{E}_{p(h_\tau|\pi)} [\mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi^{s(\tau)}) - \mathcal{I}_{\tau+1 \rightarrow T}^{h_\tau}(\pi_{h_0})] \quad (13)$$

$$\gtrsim \frac{1}{N} \left(\mathcal{L}_{\tau+1 \rightarrow T}^{h_\tau}(\pi^{s(\tau)}) - \mathcal{U}_{\tau+1 \rightarrow T}^{h_\tau}(\pi_{h_0}) \right), \quad (14)$$

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

B.3. Baselines

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

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

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

B.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 $\log y | \theta, \xi$ centered around the logarithm of the underlying model, $\mu(\theta, \xi)$.

$$\mu(\theta, \xi) = b + \sum_{k=1}^K \frac{\alpha_k}{(m + \|\theta_k - \xi\|)^2} \quad (15)$$

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.

$$\log[y \mid \theta, \xi] \sim \mathcal{N}(\log \mu(\theta, \xi), \sigma^2) \quad (16)$$

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

B.4.1. TRAINING DETAILS

The following model hyper parameters were used.

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

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

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

B.4.2. OPTIMAL τ

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 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.

B.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

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)

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 $\tau = 7$.

C. 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:

$$\log k \sim \mathcal{N}(-4.25, 1.5) \quad \alpha \sim \text{HalfNormal}(0, 2) \quad (17)$$

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:

$$V_0 = R, \quad V_1 = \frac{100}{1 + kD} \quad (18)$$

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

$$p(y = 1 | k, \alpha, R, D) = \epsilon + (1 - 2\epsilon)\Phi\left(\frac{V_1 - V_0}{\alpha}\right) \quad (19)$$

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

$$\Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} \exp -\frac{1}{2}z^2 \quad (20)$$

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.

$$D = \exp(\xi_d) \quad R = 100 \cdot \text{sigmoid}(\xi_r) \quad (21)$$

The following table gives 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 finetuning

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

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design
Publication Status	Submitted for Publication
Publication Details	Ivanova, D. R., Hedman M., Guan C., & Rainforth, T. (2024). Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design. (Accepted at ICLR 2024 DMLR Workshop, under review at NeurIPS 2024)

Student Confirmation

Student Name:	Desi R Ivanova		
Contribution to the Paper	Co-first author - Led the development of the methodology and writing of the paper - Ran experiment in Sec 6.3, contributed to the experiment in Sec 6.1 - Developed the codebase that was used for the empirical evaluations in the paper.		
Signature 	Date	19 April 2024	

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Dr Tom Rainforth		
Supervisor comments The contribution statement above is correct; Desi was the primary contributor to the project.		
Signature 	Date	19 April 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 5

CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design

This paper is published as:

Desi R Ivanova, Joel Jennings, Tom Rainforth, Cheng Zhang and Adam Foster.
CO-BED: Information-theoretic contextual optimization via Bayesian experimental
design. In *Proceedings of the 40th International Conference on Machine Learning*,
pages 14445–14464, PMLR, 2023.

CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design

Desi R. Ivanova^{1†} Joel Jennings² Tom Rainforth¹ Cheng Zhang² Adam Foster²

Abstract

We formalize the problem of contextual optimization through the lens of Bayesian experimental design and propose CO-BED—a general, model-agnostic framework for designing contextual experiments using information-theoretic principles. After formulating a suitable information-based objective, we employ black-box variational methods to simultaneously estimate it and optimize the designs in a single stochastic gradient scheme. In addition, to accommodate discrete actions within our framework, we propose leveraging continuous relaxation schemes, which can naturally be integrated into our variational objective. As a result, CO-BED provides a general and automated solution to a wide range of contextual optimization problems. We illustrate its effectiveness in a number of experiments, where CO-BED demonstrates competitive performance even when compared to bespoke, model-specific alternatives.

1. Introduction

Contextual optimization (CO) is an important problem that arises in a wide range of applications, such as drug design (Krause & Ong, 2011), nuclear fusion (Char et al., 2019; Chung et al., 2020), and robotics (Deisenroth et al., 2014; Kupcsik et al., 2017). The goal in such scenarios is to maximize a context-dependent reward function by assigning optimal actions to different contexts.

A concrete example of this problem is a personalized marketing campaign. Here different actions, such as sending marketing materials or discounts for products, are chosen based on context information such as customers’ demo-

graphics, preferences, and past engagements with the brand. The ultimate goal is to maximize revenue, but this first requires us to gather data and learn about customers’ behavior.

We consider a problem setting where we first gather data in an *experimentation stage* that consists of performing actions on a number of different contexts in parallel. At the end of the experiment, data is collected and used to inform a strategy that is then *deployed* (without additional feedback). The success of the first stage is judged on the performance of the deployed strategy: better data in the first phase should lead to better decisions and lower regret at deployment time.

Making best use of resources in the data gathering stage necessitates experimental design: we want to gather as much useful information as possible for our downstream decision-making in the deployment phase. Our first contribution is to formalize this using an information-theoretic form of Bayesian experimental design (BED, Lindley, 1956; Chaloner & Verdinelli, 1995; MacKay, 1992), thereby providing a highly principled framework for choosing designs (or actions) to be optimally informative.

Unfortunately, information-theoretic BED approaches have not previously been applied in the CO setting, or indeed with contextual information more generally. Moreover, while substantial recent progress has been made in underlying computational challenges of information-theoretic BED (Foster et al., 2019; Kleinegesse & Gutmann, 2020; Foster et al., 2020; Ivanova et al., 2021), this has generally focused on targeting information gain in model parameters, rather than the contextual optima we are interested in.

Targeting information gain in optima has separately been considered in the Bayesian optimization (BO) literature, where it is commonly referred to as entropy search (ES, Hennig & Schuler, 2012; Hernández-Lobato et al., 2014; Wang & Jegelka, 2017). However, these approaches have not been applied in contextual settings, and their usage has been heavily dependent on exploiting model-specific properties to make the required computations tractable; they cannot be directly applied in more general settings.

In this paper, we propose using an information-theoretic BED approach to CO and introduce CO-BED—a general, model-agnostic framework for designing large-scale con-

[†]Work partially conducted during an internship at Microsoft Research Cambridge. ¹Department of Statistics, University of Oxford ²Microsoft Research. Correspondence to: Desi R Ivanova <desi.ivanova@stats.ox.ac.uk>, Adam Foster <adam.e.foster@microsoft.com>.

textual experiments. We begin by formulating a suitable information-theoretic objective, the contextual max-value EIG (CMV-EIG). Importantly, CMV-EIG is *transductive*: it measures how much information an observation at one context contains about the optimum at another. As it represents a mutual information between finite-dimensional random variables, we can use black-box variational methods to simultaneously estimate CMV-EIG and optimize the designs in a single stochastic gradient scheme.

Gradient-based BED has hitherto been restricted to continuous designs, a significant limitation for contextual optimization where discrete actions are common (e.g. contextual bandits). In CO-BED, we therefore propose using the Gumbel-Softmax continuous relaxation (Maddison et al., 2016; Jang et al., 2016) to smoothly handle discrete actions.

By framing CO using BED, CO-BED does not sacrifice modelling flexibility to attain computational tractability. It instead offers a general-purpose approach that applies to a wide range of problems in seemingly disparate fields, including contextual bandits (Chu et al., 2011; Agrawal & Goyal, 2013; Langford & Zhang, 2007), contextual BO (Swersky et al., 2013; Ginsbourger et al., 2014; Pearce & Branke, 2018; Pearce et al., 2020) and structural equation models (Pearl, 2009). CO-BED naturally facilitates the design of *large batch* parallel experiments, increasingly a requirement for many applications (Groves et al., 2018; Kirsch et al., 2019; Zanette et al., 2021; Ruan et al., 2021).

We demonstrate the benefits of CO-BED in a series of experiments. Even when compared against bespoke, model-specific alternatives, we find it consistently performs on par or better, highlighting its effectiveness as a highly applicable and efficient solution. We further find it is able to scale gracefully, with effective performance maintained on a problem with a 5000 dimensional design space. Our results showcase the promising potential of CO-BED as an off-the-shelf tool for contextual optimization in various settings.

2. Background

2.1. BED with Expected Information Gain

Bayesian experimental design (BED, Lindley, 1956) is a principled model-based framework for designing optimal experiments. BED considers a Bayesian model with experimental outcomes y , controllable design $\mathbf{a}$ and latent parameter ψ with prior $p(\psi)$ and likelihood model $p(y | \psi, \mathbf{a})$. The expected information gain (EIG) about the parameters ψ is the expected reduction in entropy from the prior to the posterior distribution of ψ under an experiment with design $\mathbf{a}$:

$$\begin{aligned} \text{EIG}(\mathbf{a}) &= \mathbb{E}_{p(y|\mathbf{a})} [H[p(\psi)] - H[p(\psi | \mathbf{a}, y)]] \\ &= \mathbb{E}_{p(\psi)p(y|\psi, \mathbf{a})} \left[\log \frac{p(y | \psi, \mathbf{a})}{p(y | \mathbf{a})} \right], \end{aligned} \quad (1)$$

where $p(y | \mathbf{a}) = \mathbb{E}_{p(\psi)}[p(y | \psi, \mathbf{a})]$ is the Bayesian marginal distribution of the outcomes and is typically intractable. The EIG is equivalent to $I(\psi; y | \mathbf{a})$ —the mutual information (MI) between the parameters and the experimental outcome under the design.

A common setting, referred to as *batch*, *static* or *fixed* (Foster, 2021), is to optimize D designs $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_D)$ simultaneously to maximize the joint information objective, $\text{EIG}(\mathbf{A})$. By designing and executing informative actions, we collect a dataset $\mathcal{D} = (\mathbf{A}, \mathbf{y})$, which we use to update the model parameters in a Bayesian fashion by computing the posterior $p(\psi | \mathcal{D})$. To make inferences about any other quantity of interest, say ζ , we compute its posterior predictive distribution, $p(\zeta | \mathcal{D}) = \mathbb{E}_{p(\psi|\mathcal{D})}[p(\zeta | \mathcal{D}, \psi)]$.

2.2. Black-box MI Estimation and Optimization

Despite its highly desirable properties, estimating and maximizing the EIG (1) is notoriously difficult. This is due to its *doubly intractable* (Rainforth et al., 2018; Zheng et al., 2018) nature, which is characterized by the nested expectation structure involving a nonlinear function of an intractable term (for further details see Foster et al., 2019). To tackle this challenge, we can draw upon recent advances in self-supervised representation learning (see Poole et al., 2019, for a review) that have inspired the development of flexible, model-agnostic approaches for the *joint* estimation and optimization of information objectives. One such method is based on the InfoNCE lower bound (van den Oord et al., 2018), which has been successfully applied in a variety of model-agnostic BED contexts (Foster et al., 2020; Ivanova et al., 2021; Kleinegesse & Gutmann, 2021), and is given by

$$\begin{aligned} \text{EIG}(\mathbf{a}) &\geq \mathcal{L}(\mathbf{a}, U; L) := \\ &\mathbb{E}_{p(\psi_0:L)p(y|\mathbf{a}, \psi_0)} \left[\log \frac{\exp(U(y, \psi_0))}{\frac{1}{L+1} \sum_{\ell} \exp(U(y, \psi_{\ell}))} \right] \end{aligned} \quad (2)$$

where $\psi_0 \sim p(\psi)$ is a primary or ‘positive’ sample from the prior, $y \sim p(y | \mathbf{a}, \psi_0)$ is a realisation of the outcome under it, and $\psi_{1:L} \sim \prod_{i=1}^L p(\psi_i)$ are independent ‘contrastive’ samples. The function $U : \mathcal{Y} \times \Psi \rightarrow \mathbb{R}$ is arbitrary and commonly referred to as a *critic*. The bound becomes tight in the limit as $L \rightarrow \infty$ for the *optimal* critic $U^*(y, \psi) = p(y | \psi, \mathbf{a}) + c(y)$, where $c(y)$ can be any function that only depends on the outcome y .

If the likelihood $p(y | \psi, \mathbf{a})$ is analytically available, we can use it in (2) directly, instead of learning a critic U , thus recovering the PCE bound from Foster et al. (2020). When the likelihood is not analytically available, i.e. when we are dealing with implicit models, we parameterize U by a neural network with parameters ϕ and optimize the lower bound $\mathcal{L}(\mathbf{a}, U_{\phi}; L)$ jointly with respect to ϕ and $\mathbf{A}$, simultaneously tightening the bound and optimizing the design.

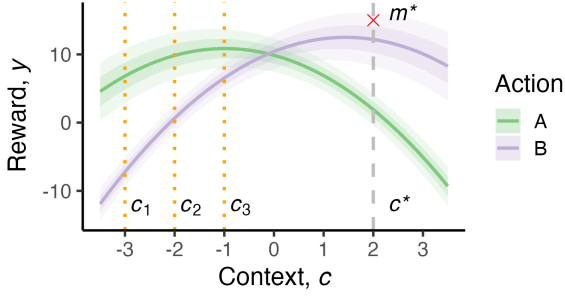


Figure 1. A stylized example. The dark (resp. light) shaded area shows our prior uncertainty about $y \mid \mathbf{a}, \mathbf{c}$ arising from uncertainty in ψ , measured by one (resp. two) st. dev. from the mean. We want to select actions in experimental contexts $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ (orange dotted lines) whose outcomes will be most informative about $\mathbf{m}^*$ (red cross) in the evaluation context $\mathbf{c}^*$ (grey dashed line).

2.3. Max-value Entropy Search (MES)

The goal of (non-contextual) Bayesian optimization is to find the global maximizer $\mathbf{a}_* = \arg \max_{\mathbf{a} \in \mathcal{A}} f(\mathbf{a})$ of some expensive, black-box function f . Max-value Entropy Search (MES, Wang & Jegelka, 2017) was proposed as a computationally efficient alternative to earlier methods, such as Entropy Search (ES, Hennig & Schuler, 2012) and Predictive Entropy Search (PES, Hernández-Lobato et al., 2014). Whilst ES and PES aim to maximize $I(\mathbf{a}_*; y \mid \mathbf{a})$ —the MI between the outcome under the action queried and the *maximizer*, MES instead uses the *maximum value*, $m = \max_{\mathbf{a} \in \mathcal{A}} f(\mathbf{a})$ and maximizes $I(m; y \mid \mathbf{a})$ w.r.t. $\mathbf{a}$. The computational efficiency of MES stems from the fact that both m and y are one-dimensional, which reduces the complexity of approximations and makes them more robust and efficient for high-dimensional problems.

Information-theoretic ES methods are popular in large batch BO, as joint MI objectives naturally handle this case. MES, like many information-theoretic approaches to BO, focused on a non-contextual Gaussian process (GP) model (Williams & Rasmussen, 2006) for the black-box function f , allowing for the use of closed-form formulae to approximate the MI.

3. Method

We introduce our approach, CO-BED: Contextual Optimization via Bayesian Experimental Design. At its core, CO-BED seeks to design a set of experiments for exploration purposes, allowing us to gather high-quality data that will lead to better decisions in the subsequent deployment stage. Code is available at <https://github.com/microsoft/co-bed>.

Problem Formulation. We extend the Bayesian model of Section 2.1 by incorporating a context vector $\mathbf{c}$ that is not under the experimenter’s control, with the likelihood becoming $p(y \mid \mathbf{a}, \mathbf{c}, \psi)$. Further, we now take y to represent a *reward* with $y \in \mathbb{R}$. We denote by $m(\mathbf{c}) = \mathbb{E}[y \mid \mathbf{a}_*, \mathbf{c}]$ the maximum value achievable in some context $\mathbf{c}$ where

$\mathbf{a}_* = \arg \max_{\mathbf{a} \in \mathcal{A}} \mathbb{E}[y \mid \mathbf{a}, \mathbf{c}]$ is the action that achieves it. Similar in spirit to MES, we wish to learn about these max-values by choosing a large batch of actions to use in an experiment. Unlike MES, we want to a) accommodate contextual information these actions are taken under, and b) make our decisions in a *transductive* manner that targets the specific contexts in which our max-values will be evaluated. Formally, given an externally provided set of *experimental* contexts $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_D)$, we seek to design a batch of actions $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_D)$ that will be maximally informative about the max rewards $\mathbf{m}^* = (m(\mathbf{c}_1^*), \dots, m(\mathbf{c}_{D^*}^*))$ for a given set of *evaluation* contexts $\mathbf{C}^* = (\mathbf{c}_1^*, \dots, \mathbf{c}_{D^*}^*)$ which are representative of contexts seen in deployment.

The contexts $\mathbf{C}$ and $\mathbf{C}^*$ are fixed but arbitrary, so they can be the same, subsets of, or distinct from each other; this is a strict generalization of the standard contextual setting $\mathbf{C} = \mathbf{C}^*$. This added flexibility can be essential in practical applications where we know about the contexts we will encounter at deployment. In the personalized marketing example, the experimental contexts $\mathbf{C}$ could represent the customers in a given city who will participate in a real-world experiment, while the evaluation contexts $\mathbf{C}^*$ may represent the customers in a whole region where the campaign will be rolled out with the updated model.

We emphasize that the goal of the design process is to obtain data that will aid in learning about the maximum rewards in the evaluation contexts, $\mathbf{m}^*$, rather than maximizing the rewards in the experimental contexts. This is illustrated in Figure 1, where choosing Action A leads to higher rewards in the experimental contexts, but these rewards will be uninformative about the value of $\mathbf{m}^*$, since this action is *a priori* known to be sub-optimal for the context of interest $\mathbf{c}^*$. This example also demonstrates that the typical EIG objective, which aims to reduce uncertainty uniformly across all model parameters ψ , is also generally sub-optimal for efficiently learning about max-values of interest. Specifically, spending experimental resources to learn the parameters associated with Action A would be ineffective and wasteful.

3.1. Contextual Max-value EIG

Following the principles of information-theoretic BED, we formulate a new objective, the contextual max-value expected information gain (CMV-EIG), for CO. Our objective focuses on the transductive gain of information about the maximum values $\mathbf{m}^*$ in the evaluation contexts $\mathbf{C}^*$ when choosing designs $\mathbf{A}$ in the experimental contexts $\mathbf{C}$:

$$\begin{aligned} \text{CMV-EIG}(\mathbf{A}; \mathbf{C}, \mathbf{C}^*) &:= \mathbb{E} \left[\log \frac{p(\mathbf{y} \mid \mathbf{m}^*, \mathbf{C}, \mathbf{A})}{p(\mathbf{y} \mid \mathbf{C}, \mathbf{A})} \right] \\ &= I(\mathbf{m}^*; \mathbf{y} \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A}). \end{aligned} \quad (3)$$

Note that this is equal to the MI between finite-dimensional random variables ($\mathbf{m}^*$ and $\mathbf{y}$). The expectation is taken over

Algorithm 1 CO-BED

EXPERIMENTATION PHASE

Input: Model $p(\psi)p(y, \mathbf{m}^*|\psi, \mathbf{c}, \mathbf{c}^*, \mathbf{a})$, initial $\mathbf{A}$ and U_ϕ , experimental contexts $\mathbf{C}$, evaluation contexts $\mathbf{C}^*$ **Output:** A batch of actions $\mathbf{A}^*$ for evaluation contexts $\mathbf{C}^*$ *Experimental design of $\mathbf{A}$*

while Computational budget not exceeded **do**
 $\triangleright$ Sample $\psi \sim p(\psi)$
 $\triangleright$ Sample $\mathbf{y}, \mathbf{m}^* \sim p(\mathbf{y}, \mathbf{m}^* | \mathbf{C}, \mathbf{C}^*, \mathbf{A})$
 $\triangleright$ Estimate $\mathcal{L}(\mathbf{A}, U_\phi; L)$ (4) using samples and update the parameters $(\mathbf{A}, \phi)$ using SGA

end*Execution of experiment, data gathering and model updating*

$\triangleright$ In parallel across $i = 1, \dots, D$, for each experimental context $\mathbf{c}_i$, apply action $\mathbf{a}_i$ obtained in previous stage, receive outcome y_i . Set $\mathbf{y} = \{y_i\}_{i=1}^D$, $\mathcal{D} = \{\mathbf{y}, \mathbf{C}, \mathbf{A}\}$
 $\triangleright$ Estimate $p(\psi | \mathcal{D})$, use it to infer optimal actions $\mathbf{A}^* = \arg \max_{\mathbf{A}' \in \mathcal{A}^{D^*}} \mathbb{E}_{p(\psi|\mathcal{D})} \mathbb{E}[y | \mathbf{A}', \mathbf{C}^*, \psi]$

DEPLOYMENT PHASE

 $\triangleright$ Apply $\mathbf{A}^*$ to obtain outcomes $\mathbf{y}^*$

the joint distribution of outcomes and quantities of interest, marginalizing out the parameters: $p(\mathbf{y}, \mathbf{m}^* | \mathbf{C}, \mathbf{C}^*, \mathbf{A}) = \mathbb{E}_{p(\psi)} [p(\mathbf{m}^* | \mathbf{C}^*, \psi)p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})]$. Similarly, the likelihood term is given by $p(\mathbf{y} | \mathbf{m}^*, \mathbf{C}, \mathbf{A}) = \mathbb{E}_{p(\psi|\mathbf{m}^*)} [p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})]$, and the marginal outcome in the denominator is $p(\mathbf{y} | \mathbf{C}, \mathbf{A}) = \mathbb{E}_{p(\psi)} [p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})]$.

By incorporating the concept of experimentation and evaluation contexts, our new objective enables more flexible and targeted experimental design by efficiently allocating resources to learn about the specific contexts of interest.

3.2. Lower bounding CMV-EIG

Our CMV-EIG objective is intractable as none of the likelihood terms involved in (3) are available analytically. To side-step this, we leverage a variational lower bound, which can be optimized with gradients using samples only. Specifically, utilizing an auxiliary *critic* function $U(\mathbf{y}, \mathbf{m}^*) \in \mathbb{R}$, we adapt the InfoNCE mutual information lower bound introduced by (van den Oord et al., 2018) and used in standard implicit-likelihood BED settings by (Ivanova et al., 2021; Kleingesse & Gutmann, 2021) to our CMV-EIG objective:

$$\mathcal{L}(\mathbf{A}, U; L) = \mathbb{E} \left[\log \frac{\exp(U(\mathbf{y}, \mathbf{m}_0^*))}{\frac{1}{L+1} \sum_{\ell} \exp(U(\mathbf{y}, \mathbf{m}_\ell^*))} \right] \quad (4)$$

$$\leq \text{CMV-EIG}(\mathbf{A}; \mathbf{C}, \mathbf{C}^*), \quad (5)$$

where the expectation is taken with respect to $p(\mathbf{y}, \mathbf{m}_0^* | \mathbf{C}, \mathbf{C}^*, \mathbf{A})p(\mathbf{m}_{1:L}^* | \mathbf{C}^*)$. The bound holds for any number of contrastive samples $L \geq 1$ and critic U , and becomes tight as $L \rightarrow \infty$ for the optimal one $U^*(\mathbf{y}, \mathbf{m}^*) = \log p(\mathbf{y} | \mathbf{m}^*, \mathbf{C}, \mathbf{A}) + c(\mathbf{y})$, where $c(\mathbf{y})$ is an arbitrary function of $\mathbf{y}$.

The key technical challenge we now face is to find a way to approximate the expectation (4) (or more specifically its gradients) in an unbiased manner. We can do that by generating joint samples $\mathbf{y}, \mathbf{m}^* \sim p(\mathbf{y}, \mathbf{m}^* | \mathbf{C}, \mathbf{C}^*, \mathbf{A}) = \mathbb{E}_{p(\psi)} [p(\mathbf{m}^* | \psi, \mathbf{C}^*)p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})]$, and contrastive max-values $\mathbf{m}_{1:L}^* \sim \prod_{\ell=1}^L p(\mathbf{m}_\ell^* | \mathbf{C}^*)$, where $p(\mathbf{m}_\ell^* | \mathbf{C}^*) = \mathbb{E}_{p(\psi)} [p(\mathbf{m}_\ell^* | \psi, \mathbf{C}^*)]$. To obtain a sample from the joint, we first sample a parameter $\psi \sim p(\psi)$, then conditionally sample the outcomes $\mathbf{y} \sim p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})$ from our model.

To sample the corresponding max-rewards $\mathbf{m}^* | \psi \sim p(\mathbf{m}^* | \psi, \mathbf{C}^*)$, we distinguish three cases. First, in certain situations, we might be able to compute $\mathbf{A}_*$ analytically, with many linear and parametric models falling into this category. For example, if $\mathbb{E}[y | \psi, \mathbf{a}, \mathbf{c}]$ depends linearly on $\mathbf{a}$, then computing $\mathbf{m}^*$ amounts to solving a linear program. Second, if $\mathcal{A}$ is discrete, we can determine the optimal actions, $\mathbf{A}_* = \arg \max_{\mathbf{A}} \mathbb{E}[y | \psi, \mathbf{A}, \mathbf{C}^*]$ by complete enumeration. This captures the majority of the contextual bandit literature. (Note that the expectation here is only over observation noise, since there is no functional uncertainty.) Third, when the previous options are not feasible, we choose a finite grid of possible actions, $\tilde{\mathcal{A}} \subset \mathcal{A}$, and get an *estimate* of $\mathbf{m}^* | \psi$ by complete enumeration over $\tilde{\mathcal{A}}$.

3.3. InfoNCE lower bound optimization

Having established how to generate joint samples, we now focus on optimizing $\mathcal{L}$ with respect to the designs and the critic. To do this in practice, we represent the critic as a neural network, U_ϕ , and optimize its parameters ϕ to improve the *tightness* of the bound; optimizing with respect to $\mathbf{A}$ improves the *quality* of the designs. We highlight that, whilst CMV-EIG resembles the MES objective (outlined in § 2.3), our approach to determining the optimal designs is quite distinct. Concretely, in its estimation procedure, MES first approximates then maximizes the MI directly. CO-BED never actually computes the MI (3) explicitly, however, provided a sufficiently flexible architecture for U_ϕ , we expect to obtain tight, high-quality estimates.

We aim to converge to the true MI maximum by jointly optimizing with respect to ϕ and designs $\mathbf{A}$ in a single stochastic gradient scheme. Whilst differentiating with respect to ϕ is straightforward, taking gradients with respect to $\mathbf{A}$ presents a technical challenge due to two reasons: 1) $\mathbf{A}$ affects the sampling of the expectation in (4); and 2) unlike network parameters, actions can be continuous or discrete.

Continuous action space. Assuming that the actions are continuous and that the experimental outcomes $y \sim p(\mathbf{y} | \psi, \mathbf{C}, \mathbf{A})$ are differentiable with respect to $\mathbf{A}$, we can form a pathwise gradient estimator (Mohamed et al., 2020) for $\nabla_{\mathbf{A}, \phi} \mathcal{L}(\mathbf{A}, U_\phi; L)$ and optimise it with standard automatic differentiation (Baydin et al., 2018; Paszke et al., 2019) and stochastic gradient schemes (Kingma & Ba, 2014).

Discrete action space. Previous gradient-based BED methods that utilize variational bounds on MI have primarily focused on fully differentiable models (see § 4 for a discussion), and avoided dealing with discrete designs. Here we propose a simple and practical way to handle discrete actions through the use of Gumbel-Softmax relaxation (Maddison et al., 2016; Jang et al., 2016). This allows us to treat the actions as continuous during the training process and apply pathwise gradients in this case as well.

Suppose we have $K \geq 2$ possible discrete actions, so that we can represent each action $(\mathbf{a}_d)_{d=1}^D$ as a one-hot vector of size K and $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_D)$ as a $D \times K$ matrix. Rather than learning $\mathbf{A}$ directly, we introduce a distribution over the actions π_α , where $\alpha \in \mathbb{R}^{D \times K}$ are trainable parameters, representing the probabilities of selecting each action in each of the D contexts. Specifically, during training, the probability of selecting action k in context d is given by:

$$\pi_{d,k} = \frac{\exp((\log \alpha_{d,k} + g_{d,k})/\tau)}{\sum_{j=1}^K \exp((\log \alpha_{d,j} + g_{d,j})/\tau)}, \quad (6)$$

where $g_{d,k} \sim \text{Gumbel}(0, 1)$ is Gumbel noise and $\tau > 0$ is a temperature hyper-parameter. We optimize the parameters α and those of the critic network ϕ jointly with SGA by sampling $\mathbf{A} \sim \pi_\alpha$ and estimating $\nabla_{\alpha, \phi} \mathcal{L}(\mathbf{A}, U_\phi; L)$. At inference time, once the policy is trained, the optimal design for experiment d in the batch is given by $\mathbf{a}_d = \arg \max_k \{\pi_{d,k}\}_{k=1}^K$.

Optimizing π_α involves making hyper-parameter choices, notably determining an appropriate value of the temperature τ , and deciding on whether or not to anneal it during training. At high temperatures, the estimates of the gradients $\nabla_{\alpha, \phi} \mathcal{L}(\mathbf{A}, U_\phi; L)$ tend to be low-variance, providing a strong learning signal for the policy to find good actions $\mathbf{A}$. Conversely, at low temperatures, gradients tend to exhibit higher variance, but π_α is closer to the discrete $\arg \max$ that we use to select the optimal action at inference time. We explore these hyper-parameter choices in an ablation study (see § 5.5), demonstrating the robustness of our framework to different temperature settings.

4. Related Work

CO-BED draws inspiration from several methods across somewhat separate fields to deliver a more general approach to contextual optimization. Our objective (3) most closely relates to MES for Bayesian optimization (Wang & Jegelka, 2017), but differs in two key ways: we focus on CO, instead of finding a single maximizer, and do not use special properties of GPs for MI estimation. Bayesian Algorithm Execution (BAX, Neiswanger et al., 2021) extends MES beyond global optima to any computable function property. Rooted in the representation learning literature, InfoNCE (4) (van den Oord et al., 2018; Wu et al., 2018) has been used

in BED for non-contextual parameter learning (Foster et al., 2020; Ivanova et al., 2021; Kleinegesse & Gutmann, 2021).

The problem we address with CO-BED relates to **contextual Bayesian optimization**, where most of the work to date has focused on iterative acquisition (i.e. batch size 1) that do not use information-theoretic criteria to choose designs. Examples of these methods include Profile Expected Improvement (PEI, Ginsbourger et al., 2014), Multi-task Thompson Sampling (MTS, Char et al., 2019) and knowledge-gradient based methods, such as LEVI, CLEVI, REVI (Pearce & Branke, 2018) and ConBO (Pearce et al., 2020). Many traditional (non-contextual) BO methods have looked at the large batch setting, using information-based criteria (Hennig & Schuler, 2012; Wang et al., 2018), and alternatives such as local penalization (LP, González et al., 2016), Multi-points Expected Improvement (q-EI, Chevalier & Ginsbourger, 2013), and the parallel knowledge-gradient (Wu & Frazier, 2016). To the best of our knowledge, the method of Groves et al. (2018), combining LEVI and LP, is the only one that considers the large batch, contextual setting and is thus the only one directly comparable to CO-BED. Sussex et al. (2022) considered BO in a structural equation model, in a non-contextual case with a known causal graph.

Our method also relates to the broad framework of **contextual bandits**. A significant portion of bandits-related research has focused on the online, linear case (Auer, 2002; Abe et al., 2003; Chu et al., 2011; Dani et al., 2008; Abbasi-Yadkori et al., 2011; Li et al., 2019). Additionally, some connections with the BO literature have been established with the introduction of Gaussian process bandit optimization methods, such as GP-UCB (Srinivas et al., 2010) and its contextual version, CGP-UCB (Krause & Ong, 2011). More recently, there has been an increased interest in the large batch setting (Han et al., 2020; Ruan et al., 2021; Zhang et al., 2021), where the goal is to achieve (some notion of) optimal regret by performing a few rounds of batched experiments. Closest to our problem set-up is the work of Zanette et al. (2021), who aim to design a single batch to collect a good dataset that is used to learn a near-optimal policy to be used at deployment time. Our approach differs from typical contextual bandits methods in that we focus on information-based criteria, instead of asymptotic regret, and do not restrict ourselves to linear models.

Our method is the first to consider *contextual* information in the the **Bayesian experimental design** framework. Using variational bounds for EIG estimation in BED for non-contextual *parameter learning* was first proposed in Foster et al. (2019). Approaches that use such bounds and optimize experimental designs using stochastic gradient procedures at the same time have subsequently been developed (Foster et al., 2020; Kleinegesse & Gutmann, 2020; 2021; Foster et al., 2021; Ivanova et al., 2021). All of these methods

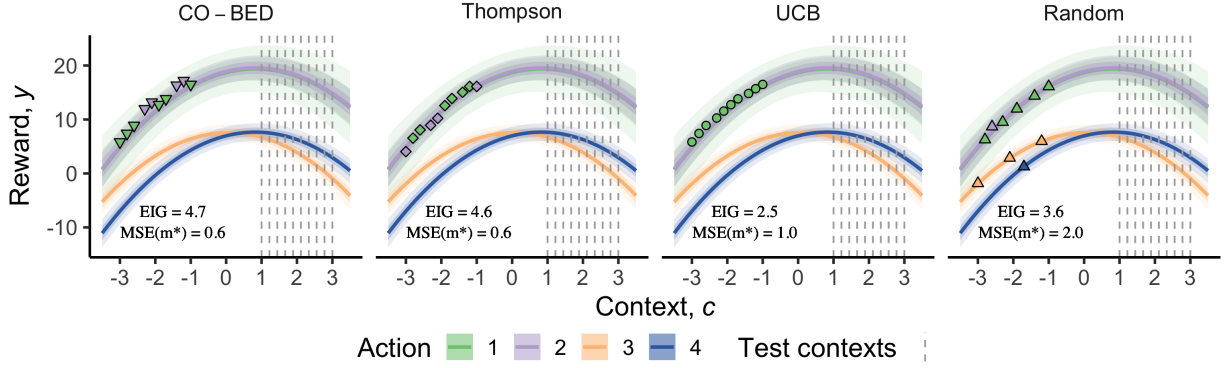


Figure 2. Discrete actions: designs and key metrics (further available in § A.2). The dashed grey lines show the evaluation contexts $\mathbf{C}^*$.

are limited in their ability to deal with *discrete designs* as they either assume fully differentiable models, or resort to gradient-free methods (Kleinegesse & Gutmann, 2020).

Finally, **Bayesian active learning** (MacKay, 1992; Houlisby et al., 2011) and Bayesian (active) causal discovery (Murphy, 2001; Tong & Koller, 2001), which can be viewed as important special cases of BED, often focus on the large batch setting, which is of particular interest for our method; notable examples of large-batch methods from the two fields include BatchBALD (Kirsch et al., 2019) and CBED (Tigas et al., 2022). Our use of an explicit evaluation context set $\mathbf{C}^*$ is akin to *transductive* active learning (MacKay, 1992; Yu et al., 2006; Reitmaier et al., 2015; Wang et al., 2021), in which one seeks data that will improve model predictions at specific inputs. The focus in active learning is to make accurate predictions, whereas CO-BED addresses the problem of choosing optimal *actions*, accepting prediction uncertainty at certain actions once they are known to be sub-optimal.

5. Empirical Evaluation

We compare the performance of **CO-BED** to a number of baselines across contextual optimization settings including contextual bandits, contextual BO, and causal structure learning. We examine continuous and discrete contexts and actions using parametric and GP-based reward models.

The baselines that we consider include model-agnostic ones, such as **random** designs, upper-confidence bound (**UCB**, Auer, 2002) and **Thompson** sampling (Thompson, 1933). We note that MTS (Char et al., 2019) reduces to pure Thompson sampling in our setting since $\mathbf{C}$ is not under the experimenter’s control. We also consider bespoke, model-specific baselines: in the experiment with GPs, we compare against **LEVI + LP** (Groves et al., 2018). For the experiment involving contextual bandits, we compare CO-BED to the Sampler-Planner (**S-P**) algorithm of (Zanette et al., 2021) and all baselines therein.

Evaluation metrics. Our evaluation metrics include the CMV-EIG itself, which we estimate by evaluating (4) with

the learnt critic and optimized design. We also consider three evaluation metrics that are useful for assessing the performance of the updated model in the deployment phase: the accuracy of inferring $\mathbf{m}^*$ and $\mathbf{A}^*$, measured by the MSE between the ground truth and the mean estimate under the posterior $p(\psi \mid \mathcal{D})$, and regret from deploying the inferred optimal actions $\mathbf{A}^*$ in the evaluation contexts $\mathbf{C}^*$. See Appendix A.1 for exact details on computing metrics.

5.1. Parametric models

We begin our empirical evaluation with two simple parametric models to ensure that our method aligns with intuition and the theory presented in the previous section. Both models have a one-dimensional, continuous context.

The first model we consider has four possible **discrete actions**, two of which are *a priori* known to be sub-optimal, whilst the other two generate rewards with the same mean, but different variances. As Figure 2 demonstrates, our method has automatically identified the intuitive optimal strategy of A/B testing only the top-performing actions. Both qualitatively and quantitatively, CO-BED performs on par with Thompson sampling and significantly outperforms the other baselines considered: the random strategy wastes resources by querying sub-optimal treatments, whilst UCB₁ only ever queries the action with higher variance.

Next, we apply CO-BED to a problem involving **continuous actions**, designing a batch of 40 experiments to learn about the max-values at 39 evaluation contexts. The Bayesian model takes the form $y = \exp(-(a - g(\psi, c))^2/h(\psi, c) - \lambda a^2) + \epsilon$, where g and h are parametric functions and ϵ is Gaussian observation noise, and we can obtain the max values in closed form. As Table 1 shows, our method outperforms the baselines on all metrics.

5.2. Gaussian Processes

We consider modelling the unknown function relating context and treatment to outcomes using a Gaussian Process (GP; Williams & Rasmussen, 2006). We explore the setting

Table 1. Continuous actions: 40D design, evenly spaced between -3.5 and 3.5, to learn max-values at 39 evaluation contexts equal to the midpoints of the grid. Random $_{\sigma}$ samples actions from $\mathcal{N}(0, \sigma)$, α in UCB $_{\alpha}$ is the confidence bound considered. Further details in § A.2.

Method	CMV-EIG $\uparrow$	MSE($\mathbf{m}^*$) $\downarrow$	MSE($\mathbf{A}$) $\downarrow$	Regret $\downarrow$
Random $_{0.2}$	5.407 $\pm$ 0.003	0.0041 $\pm$ 0.0001	0.544 $\pm$ 0.023	0.091 $\pm$ 0.002
Random $_{1.0}$	5.798 $\pm$ 0.004	0.0024 $\pm$ 0.0002	0.272 $\pm$ 0.018	0.060 $\pm$ 0.002
Random $_{2.0}$	4.960 $\pm$ 0.004	0.0042 $\pm$ 0.0002	0.450 $\pm$ 0.021	0.090 $\pm$ 0.002
UCB $_0$	5.774 $\pm$ 0.003	0.0069 $\pm$ 0.0005	0.747 $\pm$ 0.055	0.082 $\pm$ 0.002
UCB $_1$	5.876 $\pm$ 0.003	0.0030 $\pm$ 0.0002	0.338 $\pm$ 0.024	0.067 $\pm$ 0.002
UCB $_2$	5.780 $\pm$ 0.004	0.0031 $\pm$ 0.0002	0.378 $\pm$ 0.031	0.069 $\pm$ 0.002
Thompson	6.184 $\pm$ 0.004	0.0017 $\pm$ 0.0001	0.161 $\pm$ 0.007	0.051 $\pm$ 0.001
CO-BED	6.527 $\pm$ 0.003	0.0014 $\pm$ 0.0001	0.143 $\pm$ 0.018	0.044 $\pm$ 0.001

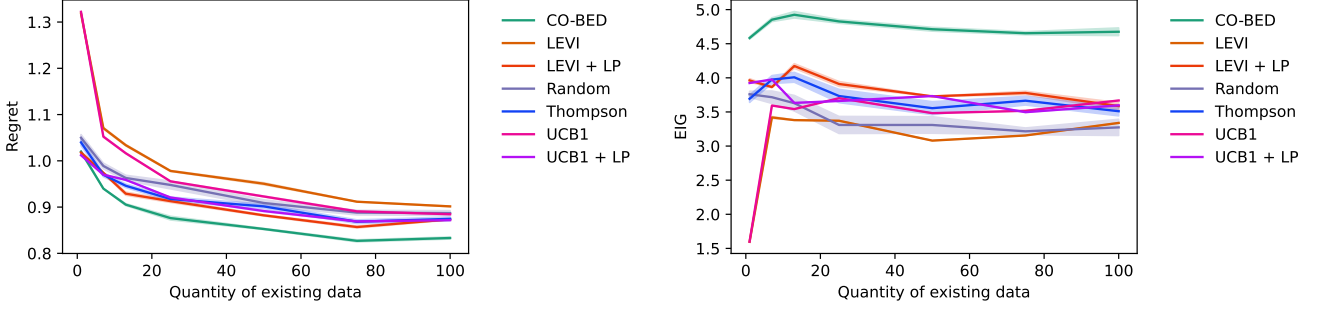


Figure 3. Results from the Gaussian Process example. Left: the regret evaluated on $\mathbf{C}^*$ after experimentation ($\downarrow$ better). Right: the lower bound on the CMV-EIG at the end of training ($\uparrow$ better). Plots show the mean ± 1 s.e. from 5 seeds. See Appendix A.3 for details.

in which experimental design begins *after* the acquisition of initial, observational data. We focus on the challenging case of confounded observational data (Greenland et al., 1999), in which the context influences the treatment chosen in the observational data. Our experimental designs must learn to counterbalance this bias. This experiment facilitates a comparison with bespoke GP methods for experimental design. We include the method of Groves et al. (2018) that combines the Local Expected Value of Improvement (LEVI; Pearce & Branke, 2018) acquisition function with Local Penalization (LP; González et al., 2016), as one of the few existing approach for batch design for contextual optimization.

Concretely, we let $\mathbf{c} \in [-1, 1]^2$, $a \in [-1, 1]$ which are inputs to $\psi : [-1, 1]^3 \rightarrow \mathbb{R}$, an unknown function modelled as $\psi \sim \mathcal{GP}(0, k)$. We use a simple Gaussian likelihood $y|\mathbf{c}, a, \psi \sim N(\psi(\mathbf{c}, a), \sigma^2)$ and a radial basis kernel, k . At design time, we condition ψ on a fraction of the 100 observational data points. At evaluation time, we sample possible ground truth functions ψ from the GP conditional on all 100 observational data (to give consistent evaluation).

Our results in Figure 3 show that CO-BED outperforms a wide range of baselines on this problem, particularly with more existing data, demonstrating that CO-BED can learn to deal with a complex prior that is defined by conditioning on confounded observational data. Baselines using LP do well, but LP’s heuristic batch design strategy does not reach the same standard as an information-optimal design.

5.3. Contextual Linear Bandits

Next, we evaluate our method on the contextual linear bandit problem described in Zanette et al. (2021), comparing CO-BED to their model-specific Sampler-Planner (S-P) algorithm, as well as the baselines considered therein—a constant strategy (Const), which always chooses action 1, largest norm strategy (Norm), which chooses the feature with the largest norm, and a random design strategy. The reward model is defined by $y = \phi(a, c)^T \psi + \epsilon$, where $\psi = (\psi_1, \dots, \psi_{20})$, is a 20-dimensional parameter vector, $a = 1, \dots, 10$ are the possible actions, $c = 1, 2, 3$ are the possible contexts, and $\phi : \mathcal{A} \times \mathcal{C} \mapsto \mathbb{R}^{20}$ is a feature map, which is assumed to be known. The problem is set up so that most actions yield zero average rewards. Specifically, only actions 1, 2, and 3 can lead to non-zero rewards in all contexts, which can be used to reduce uncertainty in certain dimensions of the parameter vector, ψ . Actions 6 and 7 give rise to non-zero features in the last dimension; however, ψ_{20} is essentially zero. All other actions yield exactly zero features. Full experiment details are given in Appendix A.4.

Figure 4 presents the results, noting that, for consistency with Zanette et al. (2021) we report the average reward, instead of regret. CO-BED outperforms the bespoke S-P algorithm at lower (< 15) design batch sizes and performs on par with it for larger, both in terms of information as well as the value of the rewards obtained during deployment. This outperformance is due to its ability to learn the information-

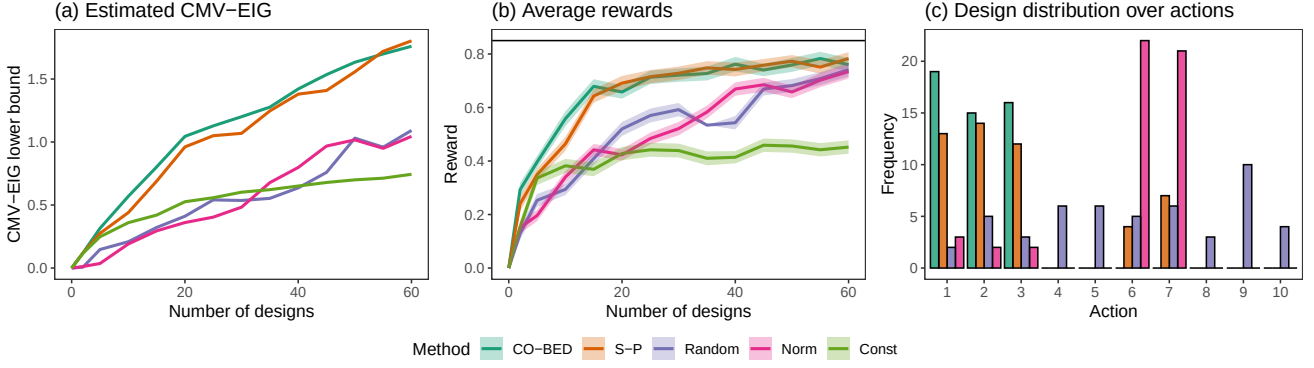


Figure 4. Linear contextual bandit results. CO-BED performs on par with the bespoke S-P algorithm both from (a) an information standpoint ($\uparrow$ better) and (b) value of the rewards obtained during deployment ($\uparrow$ better). The black horizontal line in (b) represents Supervised Learning (SL), an approximate upper bound on performance. Panel (c) shows the marginal distribution over actions of the designs obtained from each method, excluding Const.

optimal designs, as shown in Figure 4(c), specifically in its avoidance of querying actions 6 and 7, thanks to the strong close to 0 prior on ψ_{20} . Since S-P does not take prior information into account, it spends some of its experimental resources on those actions, which hurts performance when the experimental budget is low.

5.4. Unknown Causal Graph

Finally, we explore our method in the context of a structural equation model (Pearl et al., 2000; Pearl, 2009). We look at contextual optimization in a business-inspired scenario with an *unknown* causal graph. We specifically consider a binary context vector $\mathbf{c} \in \{0, 1\}^k$ which indicates which business areas a customer is active in, and treatments $\mathbf{a} \in [0, 1]^\ell$ representing investment in different promotional activities. The unobserved variable $\mathbf{r} \in \mathbb{R}^k$ indicates the revenue generated in each business area. We related these quantities using a structural equation model with a partially unknown causal graph. The unknown component of the graph describes which treatments effect which revenue streams, concretely we assume the structural equations $r_i = c_i \sum_{j=1}^\ell G_{ij} \theta_{ij} a_j$ where G_{ij} is a binary matrix and θ_{ij} are unknown linear coefficients. The total cost of treatments is simply $s = \sum_{j=1}^\ell a_j$, and the total observed profit is $y = \sum_{i=1}^k r_i - s + \epsilon$ where ϵ is Gaussian noise. The whole system is summarized in Figure 5.

This example also allows us to explore the scalability of our method. We use a fixed number of experimental contexts $D = 200$ and vary the number of possible actions ℓ up to 25, yielding designs of up to 5000 dimensions to optimize. For evaluation, we let $\mathbf{C}^*$ consist of all $2^k - 1$ non-zero binary contexts, and estimate the optimal treatments given observation $\mathbf{y}$ by fitting a Lasso (Tibshirani, 1996) due to the infeasibility of Bayesian inference in this case. See Appendix A.5 for complete details. Our results in Figure 6

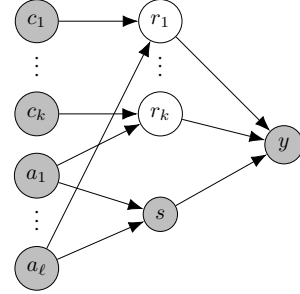


Figure 5. Causal graph considered in Section 5.4. The existence of edges from a_j to r_i is unknown. Unfilled nodes are not observed.

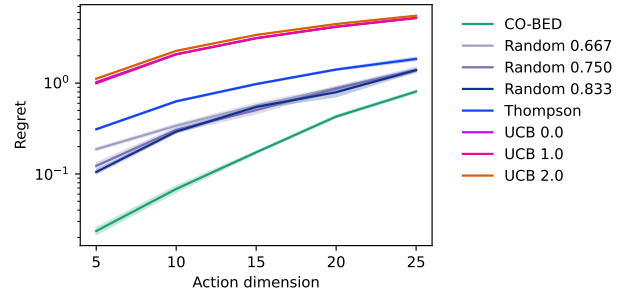


Figure 6. Results from the Unknown Causal Graph experiment showing regret on $\mathbf{C}^*$ after experimentation designed with different methods ($\downarrow$ better). Plots show mean ± 1 s.e. over 5 seeds.

show that our method is successful on this larger problem and outperforms a range of baselines. In particular, UCB baselines struggle here as they do not introduce heterogeneity between treatments.

5.5. Robustness of the Gumbel-Softmax relaxation

We perform a series of ablation studies to investigate the overall robustness of the Gumbel-Softmax relaxation scheme. We focus particularly on how different choices surrounding the temperature parameter τ can affect the performance of our framework. All experiments are performed

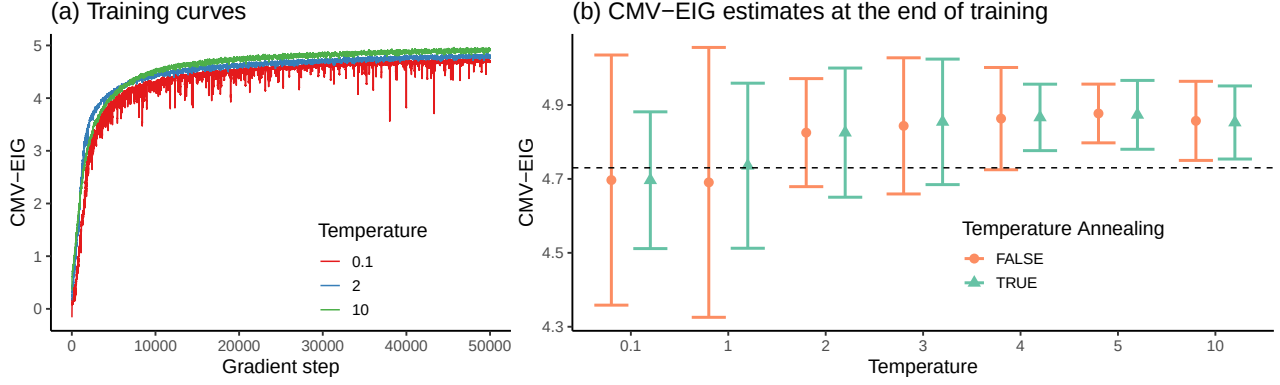


Figure 7. Robustness of the Gumbel-Softmax relaxation scheme. Panel (a) shows moving average estimates of CMV-EIG (4) (window size of 25) against gradient steps for different *fixed* values of τ (i.e. no annealing). Panel (b) presents CMV-EIG estimates computed at the end of training across 32 initializations of the training parameters (i.e. different seeds). The mean values are marked by green triangles and red dots for scenarios with and without temperature annealing, respectively. Annealing is applied every 10K steps with a factor of 0.5. Error bars indicate ± 2 st.dev., computed over the 32 seeds. The black dashed line indicates the CMV-EIG estimate reported in § 5.1 achieved by CO-BED, which was trained with temperature annealing and initial $\tau = 2.0$.

using the simple parametric model from § 5.1.

Figure 7(a) shows the training curves of the CMV-EIG lower bound (4) for three different temperature values. As anticipated, there are noticeable training instabilities at low temperature values ($\tau = 0.1$), which stem from the high variance in the estimates of the gradient $\nabla_{\alpha, \phi} \mathcal{L}(\mathbf{A}, U_{\phi}; L)$. In contrast, both moderate and high temperatures ($\tau = 2.0$ and $\tau = 10.0$) yield stable training trajectories. Importantly, despite these variations, all three settings ultimately converge towards similar values.

Next, we assess the stability across multiple training runs by optimizing $\mathcal{L}(\mathbf{A}, U_{\phi}; L)$ over 32 different seeds, showing the results in Figure 7(b). As expected, variability is larger at lower temperature values, which also tend to result in lower CMV-EIG estimates. Nevertheless, all of the CMV-EIG mean values fall within a relatively tight range between 4.7 and 4.9. However, it is worth noting that the larger CMV-EIG estimates at higher temperatures might be a byproduct of using *soft designs* during training, where we sample $\mathbf{A} \sim \pi_{\alpha}$, but use the discrete $\arg \max$ during deployment. Thus, soft training with high temperature values can potentially introduce a subtle train-test mismatch and overestimate CMV-EIG. This issue can be resolved by employing hard training or by relying on additional evaluation metrics such as regret and various accuracy metrics (as we do in the experiments section). Finally, Figure 7(b) also suggests that temperature annealing does not significantly affect performance but can help improve training stability, particularly at low temperatures.

6. Discussion

Limitations. CO-BED offers a high degree of generality as it applies to a wide range of contextual optimization problems. However, this generality comes at the cost of

increased computational cost, as it involves learning optimal actions by maximizing a lower bound on the CMV-EIG objective that requires training a (small) neural network. In many real-world applications, however, this cost is small relative to the overall cost of the experiment which may take several months to run, e.g. in marketing or medical scenario. Additionally, although common in the BO and bandits literature, future work could investigate ways to lift the assumption that y is continuous and explore more efficient ways to cheaply compute or approximate the conditional max-values $\mathbf{m}^* \mid \mathbf{C}^*, \psi$. Finally, in our experiments, we considered a scenario close to Zanette et al. (2021) involving one round of experimentation followed by one round of deployment, but there is no conceptual reason to prevent multiple, adaptive rounds.

Conclusions. We introduced CO-BED—the first method to introduce contextual aspects in the field of BED and to formally connect it to contextual optimization. By taking an information-theoretic approach, CO-BED offers a general-purpose framework that unifies seemingly disparate fields into a single cohesive framework. Our method can be end-to-end trained with gradients by employing black-box variational methods to simultaneously estimate our proposed CMV-EIG objective and optimize the designs in a single stochastic gradient scheme. Given the importance of discrete actions in optimization settings, we introduce an approach using the Gumbel-Softmax trick to handle them smoothly. We demonstrated the flexibility and effectiveness of our method across a variety of experiments, performing on par or outperforming alternative, bespoke strategies.

Acknowledgements

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

References

- Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011. 5
- Abe, N., Biermann, A. W., and Long, P. M. Reinforcement learning with immediate rewards and linear hypotheses. *Algorithmica*, 37(4):263–293, 2003. 5
- Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *International conference on machine learning*, pp. 127–135. PMLR, 2013. 2
- Annadani, Y., Rothfuss, J., Lacoste, A., Scherrer, N., Goyal, A., Bengio, Y., and Bauer, S. Variational causal networks: Approximate bayesian inference over causal structures. *arXiv preprint arXiv:2106.07635*, 2021. 18
- Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002. 5, 6
- Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18, 2018. 4
- Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 2018. 13
- Chaloner, K. and Verdinelli, I. Bayesian experimental design: A review. *Statistical Science*, pp. 273–304, 1995. 1
- Char, I., Chung, Y., Neiswanger, W., Kandasamy, K., Nelson, A. O., Boyer, M., Kolemen, E., and Schneider, J. Offline contextual bayesian optimization. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 5, 6
- Chevalier, C. and Ginsbourger, D. Fast computation of the multi-points expected improvement with applications in batch selection. In *International Conference on Learning and Intelligent Optimization*, pp. 59–69. Springer, 2013. 5
- Chu, W., Li, L., Reyzin, L., and Schapire, R. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 208–214. JMLR Workshop and Conference Proceedings, 2011. 2, 5
- Chung, Y., Char, I., Neiswanger, W., Kandasamy, K., Nelson, A. O., Boyer, M. D., Kolemen, E., and Schneider, J. Offline contextual bayesian optimization for nuclear fusion. *arXiv preprint arXiv:2001.01793*, 2020. 1
- Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic linear optimization under bandit feedback. 2008. 5
- Deisenroth, M. P., Englert, P., Peters, J., and Fox, D. Multi-task policy search for robotics. In *2014 IEEE international conference on robotics and automation (ICRA)*, pp. 3876–3881. IEEE, 2014. 1
- Foster, A., Jankowiak, M., Bingham, E., Horsfall, P., Teh, Y. W., Rainforth, T., and Goodman, N. Variational Bayesian Optimal Experimental Design. In *Advances in Neural Information Processing Systems 32*, pp. 14036–14047. Curran Associates, Inc., 2019. 1, 2, 5
- Foster, A., Jankowiak, M., O’Meara, M., Teh, Y. W., and Rainforth, T. A unified stochastic gradient approach to designing bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pp. 2959–2969. PMLR, 2020. 1, 2, 5
- Foster, A., Ivanova, D. R., Malik, I., and Rainforth, T. Deep adaptive design: Amortizing sequential bayesian experimental design. *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR 139, 2021. 5
- Foster, A. E. *Variational, Monte Carlo and Policy-Based Approaches to Bayesian Experimental Design*. PhD thesis, University of Oxford, 2021. 2
- Friedman, J., Hastie, T., and Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008. 19
- Geffner, T., Antoran, J., Foster, A., Gong, W., Ma, C., Kiciman, E., Sharma, A., Lamb, A., Kukla, M., Pawlowski, N., et al. Deep end-to-end causal inference. *arXiv preprint arXiv:2202.02195*, 2022. 18
- Ginsbourger, D., Baccou, J., Chevalier, C., Perales, F., Garland, N., and Monerie, Y. Bayesian adaptive reconstruction of profile optima and optimizers. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):490–510, 2014. 2, 5
- González, J., Dai, Z., Hennig, P., and Lawrence, N. Batch bayesian optimization via local penalization. In *Artificial intelligence and statistics*, pp. 648–657. PMLR, 2016. 5, 7
- Greenland, S., Pearl, J., and Robins, J. M. Confounding and collapsibility in causal inference. *Statistical science*, 14(1):29–46, 1999. 7
- Groves, M., Pearce, M., and Branke, J. On parallelizing multi-task Bayesian optimization. In *2018 Winter Simulation Conference (WSC)*, pp. 1993–2002. IEEE, 2018. 2, 5, 6, 7, 17

- Han, Y., Zhou, Z., Zhou, Z., Blanchet, J., Glynn, P. W., and Ye, Y. Sequential batch learning in finite-action linear contextual bandits. *arXiv preprint arXiv:2004.06321*, 2020. 5
- Hennig, P. and Schuler, C. J. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(Jun):1809–1837, 2012. 1, 3, 5
- Hernández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in neural information processing systems*, pp. 918–926, 2014. 1, 3
- Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011. 6
- Ivanova, D. R., Foster, A., Kleinegesse, S., Gutmann, M., and Rainforth, T. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. In *Advances in Neural Information Processing Systems*, volume 34, pp. 25785–25798. Curran Associates, Inc., 2021. URL <https://proceedings.neurips.cc/paper/2021/file/d811406316b669ad3d370d78b51b1d2e-Paper.pdf>. 1, 2, 4, 5
- Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 2, 5
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4, 13, 17
- Kirsch, A., Van Amersfoort, J., and Gal, Y. Batchbald: Efficient and diverse batch acquisition for deep Bayesian active learning. *Advances in neural information processing systems*, 32, 2019. 2, 6
- Kleinegesse, S. and Gutmann, M. 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, pp. 5316–5326. PMLR, 2020. 1, 5, 6
- Kleinegesse, S. and Gutmann, M. U. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds. *arXiv preprint arXiv:2105.04379*, 2021. 2, 4, 5
- Krause, A. and Ong, C. Contextual gaussian process bandit optimization. *Advances in neural information processing systems*, 24, 2011. 1, 5
- Kupcsik, A., Deisenroth, M. P., Peters, J., Loh, A. P., Vadakkepat, P., and Neumann, G. Model-based contextual policy search for data-efficient generalization of robot skills. *Artificial Intelligence*, 247:415–439, 2017. 1
- Langford, J. and Zhang, T. The epoch-greedy algorithm for contextual multi-armed bandits. *Advances in neural information processing systems*, 20(1):96–1, 2007. 2
- Li, Y., Wang, Y., and Zhou, Y. Nearly minimax-optimal regret for linearly parameterized bandits. In *Conference on Learning Theory*, pp. 2173–2174. PMLR, 2019. 5
- Lindley, D. V. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pp. 986–1005, 1956. 1, 2
- MacKay, D. J. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992. 1, 6
- Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016. 2, 5
- Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020. 4
- Murphy, K. P. Active learning of causal bayes net structure. 2001. 6
- Neiswanger, W., Wang, K. A., and Ermon, S. Bayesian algorithm execution: Estimating computable properties of black-box functions using mutual information. In *International Conference on Machine Learning*, pp. 8005–8015. PMLR, 2021. 5
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. 4, 13
- Pearce, M. and Branke, J. Continuous multi-task Bayesian optimisation with correlation. *European Journal of Operational Research*, 270(3):1074–1085, 2018. 2, 5, 7
- Pearce, M., Klaise, J., and Groves, M. Practical bayesian optimization of objectives with conditioning variables. *arXiv preprint arXiv:2002.09996*, 2020. 2, 5
- Pearl, J. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009. 2, 8, 18

- Pearl, J. et al. Models, reasoning and inference. *Cambridge, UK: Cambridge University Press*, 19, 2000. 8, 18
- Poole, B., Ozair, S., van den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, pp. 5171–5180, 2019. 2, 13, 17
- Rainforth, T., Cornish, R., Yang, H., Warrington, A., and Wood, F. On nesting monte carlo estimators. In *International Conference on Machine Learning*, pp. 4267–4276. PMLR, 2018. 2
- Reitmaier, Calma, and Sick. Transductive active learning—a new semi-supervised learning approach based on iteratively refined generative models to capture structure in data. *Information Sciences*, 2015. 6
- Ruan, Y., Yang, J., and Zhou, Y. Linear bandits with limited adaptivity and learning distributional optimal design. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 74–87, 2021. 2, 5
- Shortreed, S. M. and Ertefaie, A. Outcome-adaptive lasso: variable selection for causal inference. *Biometrics*, 73(4): 1111–1122, 2017. 19
- Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. *International Conference on Machine Learning*, 2010. 5
- Sussex, S., Makarova, A., and Krause, A. Model-based causal bayesian optimization. *arXiv preprint arXiv:2211.10257*, 2022. 5
- Swersky, K., Snoek, J., and Adams, R. P. Multi-task Bayesian optimization. *Advances in neural information processing systems*, 26, 2013. 2
- Thompson, W. R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933. 6
- Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. 8, 19
- Tigas, P., Annadani, Y., Jesson, A., Schölkopf, B., Gal, Y., and Bauer, S. Interventions, where and how? experimental design for causal models at scale. *arXiv preprint arXiv:2203.02016*, 2022. 6
- Tong, S. and Koller, D. Active learning for structure in bayesian networks. In *International joint conference on artificial intelligence*, pp. 863–869. Citeseer, 2001. 6
- van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 4, 5
- Wang, Sun, and Grosse. Beyond marginal uncertainty: how accurately can Bayesian regression models estimate posterior predictive correlations? *International Conference on Artificial Intelligence and Statistics*, 2021. 6
- Wang, Z. and Jegelka, S. Max-value entropy search for efficient bayesian optimization. In *International Conference on Machine Learning*, pp. 3627–3635. PMLR, 2017. 1, 3, 5
- Wang, Z., Gehring, C., Kohli, P., and Jegelka, S. Batched large-scale bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics*, pp. 745–754. PMLR, 2018. 5
- Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006. 3, 6
- Wu, J. and Frazier, P. The parallel knowledge gradient method for batch bayesian optimization. *Advances in neural information processing systems*, 29, 2016. 5
- Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018. 5
- Yu, Bi, and Tresp. Active learning via transductive experimental design. *International Conference on Machine Learning*, 2006. 6
- Zanette, A., Dong, K., Lee, J. N., and Brunskill, E. Design of experiments for stochastic contextual linear bandits. *Advances in Neural Information Processing Systems*, 34: 22720–22731, 2021. 2, 5, 6, 7, 9
- Zhang, Z., Ji, X., and Zhou, Y. Almost optimal batch-regret tradeoff for batch linear contextual bandits. *arXiv preprint arXiv:2110.08057*, 2021. 5
- Zheng, S., Pacheco, J., and Fisher, J. A robust approach to sequential information theoretic planning. In *International Conference on Machine Learning*, pp. 5941–5949, 2018. 2

A. Experiments

We implement all experiments in Pyro (Bingham et al., 2018), which is a probabilistic programming framework on top of PyTorch (Paszke et al., 2019). Our code will be open-sourced upon publication.

A.1. Computing evaluation metrics

Once we have an experimental design $\mathbf{A}$, we simulate the deployment phase of our main set-up (Algorithm 1) to evaluate how well experimental data using $\mathbf{A}$ enables us to perform at test time.

We begin by sampling a ground-truth parameter ψ_{true} . We then sample experimental data $\mathbf{y} \mid \psi_{\text{true}}, \mathbf{C}, \mathbf{A}$. The knowledge of the experimenter at this point is encapsulated in the posterior $p(\psi \mid \mathbf{C}, \mathbf{A}, \mathbf{y})$. Under this posterior, we can then estimate optimal actions and optimal achievable outcomes for the evaluation contexts $\mathbf{c}_1^*, \dots, \mathbf{c}_{D^*}^*$ via

$$\mathbf{a}_{\text{post},i}^* = \arg \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}_{\psi \sim p(\psi \mid \mathbf{C}, \mathbf{A}, \mathbf{y})} [\mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi]] \quad (7)$$

$$m_{\text{post},i}^* = \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}_{\psi \sim p(\psi \mid \mathbf{C}, \mathbf{A}, \mathbf{y})} [\mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi]]. \quad (8)$$

These can be compared with their counterparts under ψ_{true}

$$\mathbf{a}_{\text{true},i}^* = \arg \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi_{\text{true}}] \quad (9)$$

$$m_{\text{true},i}^* = \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi_{\text{true}}]; \quad (10)$$

giving us the ‘treatment recovery’ and ‘reward recovery’ MSEs, which are

$$L_{\text{treatment}} = \frac{1}{D^*} \sum_{i=1}^{D^*} \|\mathbf{a}_{\text{post},i}^* - \mathbf{a}_{\text{true},i}^*\|^2 \quad (11)$$

$$L_{\text{reward}} = \frac{1}{D^*} \sum_{i=1}^{D^*} |m_{\text{post},i}^* - m_{\text{true},i}^*|^2. \quad (12)$$

Finally, we evaluate the regret under ψ_{true} from acting with $\mathbf{A}_{\text{post}}^*$ as opposed to $\mathbf{A}_{\text{true}}^*$. This is defined as

$$r_i = m_{\text{true},i}^* - \mathbb{E}[y \mid \mathbf{a}_{\text{post},i}^*, \mathbf{c}_i^*, \psi_{\text{true}}] \quad r = \frac{1}{D^*} \sum_i r_i. \quad (13)$$

To give a less biased evaluation, this procedure is repeated for several thousand ground truth parameters ψ_{true} and the results are averaged. The exact number of true models considered is given in the following sections.

A.2. Parametric models

Training details All experiment baselines ran for 50K gradient steps, using a batch size of 2048. We used the Adam optimiser (Kingma & Ba, 2014) with an initial learning rate of 0.001 and exponential learning rate annealing with a coefficient of 0.96 applied every 1000 steps. We used a separable critic architecture (Poole et al., 2019) with simple MLP encoders with ReLU activations and 32 output units.

For the discrete treatment example: we added *batch norm* to the critic architecture, which helped to stabilise the optimisation. We had one hidden layer of size 512. Additionally, for the Gumbel–Softmax policy, we started with a temperature $\tau = 2.0$ and `hard=False` constraint. We applied temperature annealing every 10K steps with a factor of 0.5. We switch to `hard=True` in the last 10K steps of training.

For the continuous treatment example: We used MLPs with hidden layers of sizes $[\text{design dimension} \times 2; 412; 256]$ and 32 output units.

Note: In order to evaluate the EIG of various baselines, we train a critic network for each one of them with the same hyperparameters as above.

Table 2. Discrete treatments: evaluation metrics of 10D design

Method	EIG estimate	MSE($\mathbf{m}^*$)	Hit rate($\mathbf{A}$)	Regret
UCB _{0.0}	1.735 $\pm$ 0.005	2.541 $\pm$ 0.104	0.513 $\pm$ 0.01	1.170 $\pm$ 0.036
UCB _{1.0}	2.514 $\pm$ 0.006	1.003 $\pm$ 0.043	0.496 $\pm$ 0.01	1.119 $\pm$ 0.035
UCB _{2.0}	2.504 $\pm$ 0.006	0.965 $\pm$ 0.045	0.497 $\pm$ 0.01	1.169 $\pm$ 0.037
Thompson	4.607 $\pm$ 0.007	0.620 $\pm$ 0.024	0.498 $\pm$ 0.01	1.112 $\pm$ 0.035
Random	3.573 $\pm$ 0.006	1.953 $\pm$ 0.070	0.503 $\pm$ 0.01	1.150 $\pm$ 0.036
Ours	4.729 $\pm$ 0.009	0.594 $\pm$ 0.025	0.501 $\pm$ 0.01	1.152 $\pm$ 0.035

Posterior inference details After completing the training stage of our method (Algorithm 1), we need to deploy the learnt optimal designs in the real world in order to obtain rewards y . This experimental data is then used to fit a posterior $p(\psi|\mathcal{D})$.

There are many ways to do the posterior inference and the quality of the results will crucially depend on the accuracy of the fitted posteriors. In both of our examples and for all baselines we use Pyro’s self-normalised importance sampling (SNIS). Samples from this posterior are used for the evaluation metrics.

We validate the accuracy of estimated posteriors by running various sanity checks, including diagnostic plots such as Figure 8, showing the standard deviation of our posterior mean estimate (a measure of uncertainty about the parameter) and L_2 error to the true parameter. The red line shows the rolling mean over 200 points of the latter, and the grey band—the 2 standard deviations. For this plot we used the example of the continuous action with $D = 20$ experimental contexts.

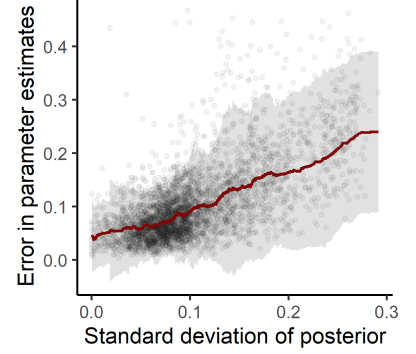


Figure 8. Posterior checks

Evaluation metrics details As discussed in the main text, we evaluate how well we can estimate $\mathbf{m}^*$ by sampling a ground truth ψ from the prior and obtaining a corresponding ground truth $\tilde{\mathbf{m}}^*$. We approximate the max-values $\mathbf{m}^*$ empirically using 2000 posterior samples of ψ . We similarly estimate ψ using 2000 posterior samples. We define the optimal action under the posterior model to be the average (with respect to that posterior) optimal action when actions are continuous or UCB₀ when discrete. Finally, the regret is computed as the average difference between the true max value (from the true environment and the true optimal action) and the one obtained by applying the estimated optimal action. We used 4000 (resp. 2000) true environment realisation for the continuous (resp. discrete) example.

A.2.1. DISCRETE ACTIONS EXAMPLE

Model We first give details about the toy model we consider in Figure 2. Each of the four treatments $\mathbf{a} = 1, 2, 3, 4$ is a random function with two parameters $\psi_k = (\psi_{k,1}, \psi_{k,2})$ with the following Gaussian priors (parameterised by mean and covariance matrix):

$$\psi_1 \sim \mathcal{N}\left(\begin{pmatrix} 5.00 \\ 15.0 \end{pmatrix}, \begin{pmatrix} 9.00 & 0 \\ 0 & 9.00 \end{pmatrix}\right) \quad \psi_2 \sim \mathcal{N}\left(\begin{pmatrix} 5.00 \\ 15.0 \end{pmatrix}, \begin{pmatrix} 2.25 & 0 \\ 0 & 2.25 \end{pmatrix}\right) \quad (14)$$

$$\psi_3 \sim \mathcal{N}\left(\begin{pmatrix} -2.0 \\ -1.0 \end{pmatrix}, \begin{pmatrix} 1.21 & 0 \\ 0 & 1.21 \end{pmatrix}\right) \quad \psi_4 \sim \mathcal{N}\left(\begin{pmatrix} -7.0 \\ 3.0 \end{pmatrix}, \begin{pmatrix} 1.21 & 0 \\ 0 & 1.21 \end{pmatrix}\right) \quad (15)$$

and reward (outcome) likelihoods:

$$y|\mathbf{c}, \mathbf{a}, \psi \sim \mathcal{N}(f(\mathbf{c}, \mathbf{a}, \psi), 0.1) \quad (16)$$

$$f(\mathbf{c}, \mathbf{a}, \psi) = -\mathbf{c}^2 + \beta(\mathbf{a}, \psi)\mathbf{c} + \gamma(\mathbf{a}, \psi) \quad (17)$$

$$\gamma = (\psi_{\mathbf{a},1} + \psi_{\mathbf{a},2} + 18)/2 \quad (18)$$

$$\beta = (\psi_{\mathbf{a},2} - \gamma + 9)/3 \quad (19)$$

Intuition about the parameterisation: The first component of each ψ_i defines the mean reward at context $\mathbf{c} = -3$, while the second one defines the mean reward at context $\mathbf{c} = 3$. The reward is then the quadratic equation that passes through those

Table 3. Discrete treatments example: 10D design, stability across training seeds

Method	EIG estimate	MSE($\mathbf{m}^*$)	Hit rate($\mathbf{A}$)	Regret
UCB _{0.0}	1.740 ± 0.003	2.709 ± 0.058	0.500 ± 0.005	1.150 ± 0.017
UCB _{1.0}	2.508 ± 0.002	0.993 ± 0.016	0.498 ± 0.004	1.140 ± 0.007
UCB _{2.0}	2.505 ± 0.006	0.991 ± 0.023	0.497 ± 0.003	1.145 ± 0.015
Thompson	4.536 ± 0.173	0.773 ± 0.127	0.500 ± 0.003	1.148 ± 0.018
Random	3.573 ± 0.333	2.369 ± 0.382	0.502 ± 0.003	1.166 ± 0.008
Ours	4.769 ± 0.048	0.628 ± 0.025	0.502 ± 0.005	1.160 ± 0.021

Table 4. Continuous actions: 40D design training stability. Mean and standard error are reported across 6 different training seeds.

Method	EIG estimate	MSE($\mathbf{m}^*$)	MSE($\mathbf{A}$)	Regret
Random _{0.2}	5.548 ± 0.044	0.0037 ± 0.0002	0.451 ± 0.033	0.083 ± 0.004
Random _{1.0}	5.654 ± 0.128	0.0031 ± 0.0004	0.343 ± 0.044	0.069 ± 0.006
Random _{2.0}	5.118 ± 0.163	0.0045 ± 0.0003	0.498 ± 0.032	0.086 ± 0.004
UCB _{0.0}	5.768 ± 0.002	0.0066 ± 0.0002	0.729 ± 0.022	0.082 ± 0.001
UCB _{1.0}	5.892 ± 0.006	0.0031 ± 0.0001	0.354 ± 0.013	0.068 ± 0.001
UCB _{2.0}	5.797 ± 0.004	0.0030 ± 0.0001	0.343 ± 0.011	0.071 ± 0.001
Thompson	6.184 ± 0.004	0.0017 ± 0.0001	0.161 ± 0.007	0.051 ± 0.001
Ours	6.538 ± 0.008	0.0013 ± 0.0001	0.131 ± 0.006	0.042 ± 0.0001

points and the leading coefficient is equal to -1 .

Experimental and evaluation contexts We use experimental and evaluation contexts of the same sizes. The experimental context, $\mathbf{c}$ is an equally spaced grid of size 10 between -3 and -1 . We set the evaluation context $\mathbf{c}^* = -\mathbf{c}$. Figure 2 in the main text visually illustrates this: the x -axis of the points in each plot are the experimental contexts, while the dashed gray lines are the evaluation contexts.

Further results Table 2 shows all the evaluation metrics for the discrete treatment example from the main text. Our method achieves substantially higher EIG and lower MSE for estimating the max-rewards. On all other metrics, all methods perform similarly. This is to be expected since Treatments 1 and 2 have exactly the same means and due to the way the model was parameterised (by the value of the quadratic at contexts 3 and -3), the probability of the optimal treatment being 1 or 2 is exactly 50% (the hit rate all baselines achieve). Note that UCB₁ and UCB₂ achieve statistically identical results, which is expected given they select the same designs.

Training stability We perform our method with the same hyperparameters but different training seeds and report the mean and standard error in Table 3.

A.2.2. CONTINUOUS TREATMENT EXAMPLE

Model For the continuous treatment example we use the following model:

$$\text{Prior: } \psi = (\psi_0, \psi_1, \psi_2, \psi_3), \quad \psi_i \sim \text{Uniform}[0.1, 1.1] \text{ iid} \quad (20)$$

$$\text{Likelihood: } y|\mathbf{c}, \mathbf{a}, \psi \sim \mathcal{N}(f(\psi, \mathbf{a}, \mathbf{c}), \sigma^2), \quad (21)$$

where

$$f(\psi, \mathbf{a}, \mathbf{c}) = \exp\left(-\frac{(a - g(\psi, \mathbf{c}))^2}{h(\psi, \mathbf{c})} - \lambda a^2\right) \quad g(\psi, \mathbf{c}) = \psi_0 + \psi_1 \mathbf{c} + \psi_2 \mathbf{c}^2 \quad h(\psi, \mathbf{c}) = \psi_3. \quad (22)$$

The parameter λ is a cost weight, we set $\lambda = 0.1$ in our experiments.

Table 5. Continuous actions: 20D design to learn about 19 evaluation contexts.

Method	EIG estimate	MSE($\mathbf{m}^*$)	MSE($\mathbf{A}$)	Regret
Random _{0.2}	4.262 ± 0.004	0.0086 ± 0.0003	1.046 ± 0.041	0.120 ± 0.002
Random _{1.0}	4.264 ± 0.004	0.0068 ± 0.0003	0.799 ± 0.033	0.114 ± 0.002
Random _{2.0}	4.116 ± 0.003	0.0083 ± 0.0003	1.002 ± 0.044	0.127 ± 0.003
UCB _{0.0}	5.093 ± 0.004	0.0074 ± 0.0004	0.800 ± 0.047	0.097 ± 0.002
UCB _{1.0}	5.040 ± 0.004	0.0072 ± 0.0004	0.764 ± 0.041	0.097 ± 0.002
UCB _{2.0}	5.038 ± 0.004	0.0048 ± 0.0003	0.573 ± 0.033	0.086 ± 0.002
Thompson	4.924 ± 0.045	0.0055 ± 0.0004	0.547 ± 0.054	0.093 ± 0.003
Ours	5.642 ± 0.003	0.0034 ± 0.0002	0.065 ± 0.002	0.034 ± 0.027

Table 6. Continuous treatment example: 60D design to learn about 59 evaluation contexts.

Method	EIG estimate	MSE($\mathbf{m}^*$)	MSE($\mathbf{A}$)	Regret
Random _{0.2}	6.033 ± 0.003	0.0026 ± 0.0001	0.307 ± 0.019	0.068 ± 0.002
Random _{1.0}	5.877 ± 0.004	0.0025 ± 0.0002	0.310 ± 0.023	0.064 ± 0.002
Random _{2.0}	6.153 ± 0.003	0.0022 ± 0.0002	0.226 ± 0.019	0.055 ± 0.002
UCB _{0.0}	6.106 ± 0.003	0.0056 ± 0.0004	0.586 ± 0.045	0.077 ± 0.002
UCB _{1.0}	6.200 ± 0.003	0.0027 ± 0.0003	0.305 ± 0.028	0.063 ± 0.002
UCB _{2.0}	6.234 ± 0.003	0.0024 ± 0.0002	0.252 ± 0.024	0.064 ± 0.002
Thompson	6.656 ± 0.018	0.0012 ± 0.0001	0.105 ± 0.008	0.043 ± 0.001
Ours	6.932 ± 0.003	0.0007 ± 0.0001	0.069 ± 0.009	0.033 ± 0.001

Experimental and evaluation contexts The experimental context, $\mathbf{c}$ is an equally spaced grid of size $D = 40$ (or 20 or 60 in Further Results below) between -3.5 and 3.5 . The evaluation context $\mathbf{c}^*$ is of size $D^* = D - 1$ and consists of the midpoints of the experimental context.

Baselines Since we have a continuous treatment, for the random baseline we consider sampling designs at random from $\mathcal{N}(0, 0.2)$, $\mathcal{N}(0, 1)$ or $\mathcal{N}(0, 2)$, which we denote by Random_{0.2}, Random₁ and Random₂, respectively.

Training stability We perform our method with the same hyperparameters but different training seeds and report the mean and standard error in Table 4.

Further results We report the results of the same experiment, but with a smaller and larger batch sizes of experimental and evaluation contexts. Table 5 shows results for an experimental batch size of 20 contexts to learn about 19 evaluation contexts. Finally, Table 6 shows results for an experimental batch size of 60 contexts to learn about 59 evaluation contexts.

A.3. Gaussian Processes

We take $\mathbf{c} \in [-1, 1]^2$ and $a \in [-1, 1]$. We consider a GP $\psi \sim \mathcal{GP}(0, k)$ where k is a radial basis kernel with length-scale $\frac{1}{3}$. Observations are sampled as $y|\mathbf{c}, a, \psi \sim N(\psi(\mathbf{c}, a), \sigma^2)$.

Formally, the confounding bias in observational data arises from the causal graph in Figure 9. Concretely, we created 100 initial observational data points; this data was then held fixed across all experiment runs and seeds. The data was created by sampling $c_i \stackrel{\text{i.i.d.}}{\sim} \text{Unif}(-1, 1)$. To create a confounded dataset with $\mathbf{c}$ acting as a confounder, we take $a = \text{sign}(c_1)\text{Unif}(0.8, 1)$. Finally, we let $y = 1 + \sin(\pi(c_1 - c_2)) - (a - \sin(\pi(c_1 + c_2)))^2$. Note, this function is *not* used for evaluation, instead we sample possible ground truth functions from the GP conditioned on all 100 observational data points. This allows us to validate the robustness of our method to different ground truth functions, and is in keeping with our other experiments. For the purely random existing data, we resample using exactly the same procedure, except $a \sim \text{Unif}(-1, 1)$ in this case.

The experimental context $\mathbf{C}$ was an evenly spaced 7×7 grid with corners at $(\pm 1, \pm 1)$. The evaluation context $\mathbf{C}^*$ was an evenly spaced 4×4 grid with corners at $(\pm 0.8, \pm 0.8)$.

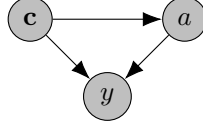


Figure 9. The form of the causal graph that generates observational data for Section 5.2.

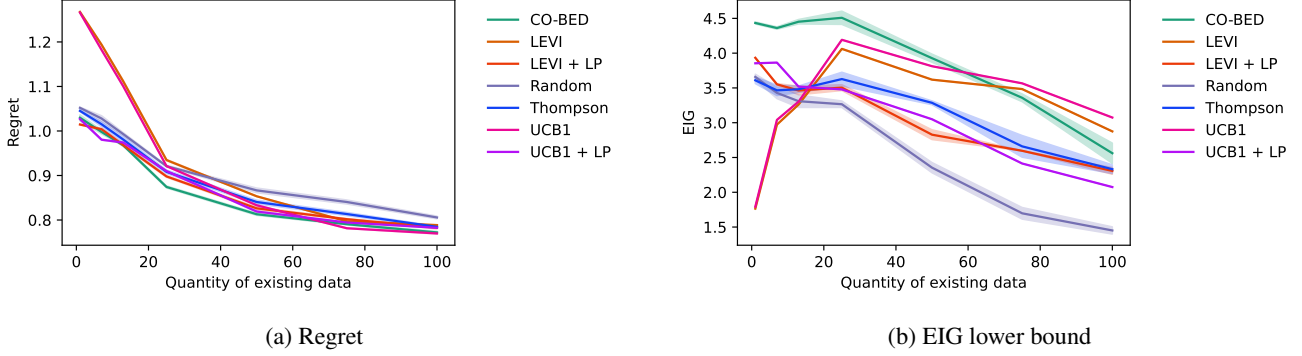


Figure 10. Results from the Gaussian Process example with purely random existing data. We show the mean ± 1 s.e. from 5 random seeds.

Since it is not possible to sample the infinite dimensional ψ , we instead take joint samples of ψ evaluated at $(\mathbf{c}_i, a_i)_{i=1}^D, (\mathbf{c}_j^*, g_k)_{j=1, k=1}^{D^*, G}$ where $g_1, \dots, g_G$ is a uniform grid covering $[-1, 1]$. From this set of joint samples, we compute each $(m_j^*)_{j=1}^{D^*}$ by maximising over the grid. Sampling of the multivariate Gaussian admits pathwise derivatives, which we utilise to optimise the design.

For the random baseline, we sample $a \sim \text{Unif}(-1, 1)$. For LP, we follow Groves et al. (2018) and use a penalization function of the form $1 - k^0(a, a')$. The kernel k^0 is chosen to be a RBF kernel with the same length-scale as the kernel of the GP model itself. We apply the same penalization scheme for the UCB + LP baseline. Table 7 details all the settings used in this experiment.

We also performed the same experiment, but with observational $\mathbf{c}, a$ sampled independently and uniformly (no confounding bias). In this case, the benefits of CO-BED are reduced (Figure 10), although it still does on par with the best of the baselines. Likely, this is because simply ‘spreading out’ designs is a good approach in this case.

A.4. Contextual Linear Bandits

The random features $\phi(a, c)$ are sampled from $\mathcal{N}(0, \Sigma_{a,c})$, where the covariance matrices are all diagonal and of the form $\Sigma_{a,c} = \text{diag}(10^{-9}, \dots, 10^{-9}, 1, 10^{-9}, \dots, 10^{-9})$, with the position of 1 specified as follows:

- Case $c = 1$: $(\Sigma_{1,1})_{11} = 1, (\Sigma_{2,1})_{22} = 1, (\Sigma_{3,1})_{33} = 1,$
- Case $c = 2$: $(\Sigma_{1,2})_{44} = 1, (\Sigma_{2,2})_{11} = 1, (\Sigma_{3,2})_{55} = 1,$
- Case $c = 3$: $(\Sigma_{1,3})_{66} = 1, (\Sigma_{2,3})_{77} = 1, (\Sigma_{3,3})_{11} = 1,$

We define the following prior on the parameters ψ : $\psi_{1:19} \sim \mathcal{N}(0, 1.0)$ iid, and $\psi_{20} \sim \mathcal{N}(0, 0.1)$.

Experimental and evaluation contexts The experimental contexts $\mathbf{C}$ are sampled uniformly from $\{1, 2, 3\}$, whilst $\mathbf{C}^* = (1, 2, 3)$. We varied the design dimension $2, 5, 10, 15, \dots, 60$.

Training details All experiments baselines ran for 100K gradient steps, using a batch size of 1024. We used the Adam optimiser (Kingma & Ba, 2014) with initial learning rate $1e^{-3}$ and exponential learning rate annealing with coefficient 0.96 applied every 1000 steps. We used a separable critic architecture (Poole et al., 2019) with simple MLP encoders with ReLU activations hidden units determined by the size of the design. Concretely, we use MLPs with sizes `[input_dim,`

Table 7. Parameter settings for the Gaussian Process experiment.

Parameter	Value
Intrinsic context dimension	2
Intrinsic treatment dimension	1
Experimental batch size, D	49
Evaluation batch size, D^*	16
RBF kernel length-scale	$\frac{1}{3}$
Observation noise σ	0.1
Treatment grid size	128
Number of training steps	50000
Initial learning rate	0.001
Learning rate decay factor	0.96
Training batch size	2048
Critic encoding dimension	32
Critic hidden dimension	256
Number of ground truth evaluation functions	3000

$2 \times \text{input_dim}, 4 \times \text{input_dim}, \text{input_dim}]$, where input_dim is equal to D (resp. D^*) for encoding the outcomes $\mathbf{y}$ (resp. max-values $\mathbf{m}^*$).

We added *batch norm* to the critic architecture, which helped to stabilize the optimisation. Additionally, for the Gumbel–Softmax policy, we started with a temperature $\tau = 5.0$ and `hard=False` constraint. We applied temperature annealing every 20K steps with a factor 0.5. We switch to `hard=True` in the last 20K steps of training.

Note: In order to evaluate the EIG of various baselines, we train a critic network for each one of them with the same hyperparameters as above.

A.5. Unknown Causal Graph

Structural equation modelling (Pearl et al., 2000; Pearl, 2009) is a mainstay of causal reasoning in statistics. The causal assumptions of this experiment are captured explicitly in Figure 5. This causal graph captures the intuition that $\mathbf{c}$ represents contexts that cannot be changed in the short run and $\mathbf{a}$ represents actions that are directly manipulated.

We consider a binary context vector $\mathbf{c} \in \{0, 1\}^k$ indicating which business areas a customer is active in, and treatments $\mathbf{a} \in [0, 1]^\ell$ representing investment in different promotional activities. The *unobserved* variable $\mathbf{r} \in \mathbb{R}^k$ indicates the revenue generated in each business area. The unknown component of the causal graph relates to which treatments effect which revenue streams, with $r_i = c_i \sum_{j=1}^\ell G_{ij} \theta_{ij} a_j$ where G_{ij} is a binary causal graph and θ_{ij} are linear coefficients. The latent variables of the model are therefore $\psi = \{G, \theta\}$, each a matrix of shape $k \times \ell$. In the prior, we sample each component of G independently from $\text{Bernoulli}(3/2k)$ and each component of $\theta \stackrel{\text{i.i.d.}}{\sim} \text{HalfNormal}(1)$. Note that, given our set-up, any sample of G results in the overall causal graph being acyclic, side-stepping some of the complexities of learning distributions over causal graphs more generally (Annadani et al., 2021; Geffner et al., 2022). The total cost of treatments is simply $s = \sum_{j=1}^\ell a_j$, and the total profit is $y = \sum_{i=1}^k r_i - s + \epsilon$ where ϵ is sampled $\epsilon \sim N(0.25^2)$.

At design time, we create a random experimental context $\mathbf{C} \in \{0, 1\}^{D \times k}$. We fix $D = 200, k = 8$ and sample each component of the context from $\text{Bernoulli}(0.5)$. This context is sampled once and fixed across seeds and baselines, to focus differences on quality of experimental design.

For CO-BED, we represent $\mathbf{a}$ in logit space, and use an initialization of $N(0, 1)$. We can compute conditional maximum rewards $m^* \mid \mathbf{c}^*, G, \theta$ using the formula

$$u_j := -1 + \sum_{i=1}^k c_i G_{ij} \theta_{ij}, \quad a_j^* = \begin{cases} 1 & \text{if } u_j > 0, \\ 0 & \text{otherwise} \end{cases}, \quad m^* = \sum_{j=1}^\ell \left(\sum_{i=1}^k c_i G_{ij} \theta_{ij} a_j^* \right) - a_j^*. \quad (23)$$

For the random baselines, we restricted ourselves to designs placed at the extrema, i.e. $\mathbf{a} \in \{0, 1\}^\ell$. Since the functional relationships in the model are linear, using only extreme values is likely to substantially improve the quality of the baseline.

Table 8. Parameter settings for the Unknown Causal Graph experiment.

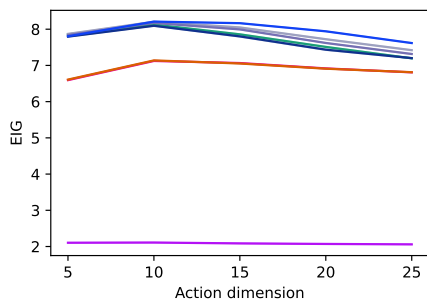
Parameter	Value
Intrinsic context dimension, k	8
Intrinsic treatment dimension, ℓ	5, 10, 15, 20, 25
Experimental batch size, D	200
Experimental context sampling probability	0.5
Evaluation batch size, D^*	255
Observation noise scale	0.25
Graph prior probability	$\frac{3}{16}$
Linear coefficient prior	HalfNormal(1)
Random baseline probability p	0.667, 0.75, 0.833
Number of training steps	400000
Initial learning rate (critic)	3×10^{-6}
Initial learning rate (design)	3×10^{-5}
Learning rate decay factor	0.998
Training batch size	4096
Critic encoding dimension	256
Critic hidden dimension	512
Number of ground truth evaluation functions	10000

We sampled the baseline design components independently from Bernoulli(p) for various values of p . The UCB designs were computed by first calculating upper confidence bounds on each entry of $G \odot \theta$ (here, $\odot$ indicates the Hadamard, or element-wise, product).

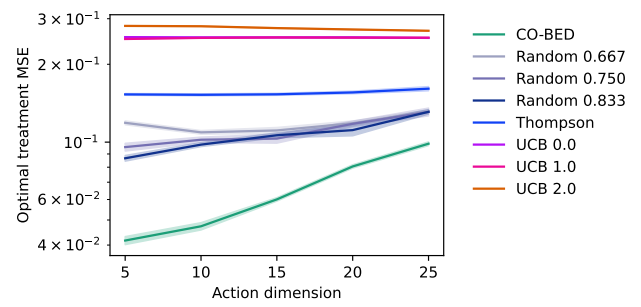
At evaluation time, we created a systematic evaluation context $\mathbf{C} \in \mathbb{R}^{D^* \times k}$ that consists of *all* $D^* = 2^k - 1$ non-zero binary context vectors of length k . This means that our evaluation is ‘comprehensive’ in the sense that it covers all possible contexts that might be observed. With our choice $k = 8$ we have $D^* = 255$, which is slightly larger than the total number of experiments $D = 200$.

Our standard approach for evaluation is to first sample a ground truth ψ^* from the prior, sample experimental outcomes $\mathbf{c} \sim p(\mathbf{y}|\mathbf{C}, \mathbf{A}, \psi^*)$, and then compute the posterior $p(\psi|\mathbf{y}, \mathbf{C}, \mathbf{A})$. However, in this case, computing this posterior constitutes solving a partial causal discovery problem on an adjacency matrix of size $k \times \ell$; doing this accurately is an area of ongoing active research. Instead, we substitute the posterior calculation for a point estimate. We begin by observing that $y + s = \sum_{i,j} c_i G_{ij} \theta_{ij} a_j + \epsilon$ can be interpreted as a linear model with coefficient equal to the pointwise product $G \odot \theta$ and covariates given by the outer product of $\mathbf{a}$ and $\mathbf{c}$. We therefore estimate $G \odot \theta$ by fitting a Lasso (Tibshirani, 1996) to the data that was sampled under ψ^* . The Lasso has a long history in causal discovery, and is considered a robust approach to estimating the causal parents (Friedman et al., 2008; Shortreed & Ertefaie, 2017). We select the Lasso penalty weight α using cross validation independently for each run. Our results appear to accord very well with the ground truth graphs and optimal actions, particularly at lower dimensions, indicating that our approach to approximate causal discovery is suitable in this case.

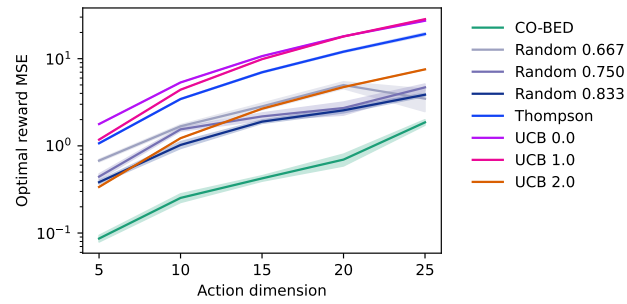
Figure 11 shows additional metrics from our experiment. Interestingly, we see that CO-BED does *not* outperform other methods on the EIG metric, despite this being the objective that is directly optimised. We believe that this finding is related to the EIG objective for these large scale experiments being near to its maximum value of $\log(4096) = 8.318$. It is not possible to exceed this bound without increasing the batch size further. Secondly, for the baselines, the critic can fully adapt to a fixed design throughout 400000 training iterations, whereas for CO-BED, the critic has to adapt to a design that changes during training, and is therefore likely to improve further with yet longer training. This experiment shows convincingly that the InfoNCE objective can give good training gradients for experimental designs *even when it is close to saturation at* $\log(\text{batch size})$. Finally, Table 8 details the settings used in our experiment.



(a) CMV-EIG



(b) Optimal treatment MSE



(c) Optimal reward MSE

Figure 11. Additional metrics from the Unknown Causal Graph experiment. Plots show the mean ± 1 s.e. from 5 seeds.

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design
Publication Status	Published
Publication Details	Ivanova, D. R., Jennings, J., Rainforth, T., Zhang, C., & Foster, A. (2023, July). CO-BED: Information-theoretic contextual optimization via Bayesian experimental design. In <i>International Conference on Machine Learning</i> (pp. 14445-14464). PMLR.

Student Confirmation

Student Name:	Desi R Ivanova		
Contribution to the Paper	First author - Led the writing of the paper and significantly contributed to the methodology - Ran experiments in Sec 5.1 and 5.3 - Developed the codebase that was used for the empirical evaluations in the paper.		
Signature 	Date	18 April 2024	

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Dr Tom Rainforth		
Supervisor comments The contribution statement above is correct; Desi was the primary contributor to the project.		
Signature 	Date	19 April 2024

This completed form should be included in the thesis, at the end of the relevant chapter.

Chapter 6

Discussion

6.1 Frequently asked questions (FAQs)

In this section, I aim to answer some of the frequently asked questions related to Chapters 2–5.

What data do you use to train the policy? Or (more educated) versions of it, such as “*How do you generate training data to learn the policy?*” or “*What distribution do the simulated experiments come from?*”, is probably the most frequently asked question.

Unlike traditional supervised learning, where one typically begins with a labeled dataset, in BED, we start with no data at all. Instead, we rely on our model (or simulator) to generate hypothetical trajectories that then serve as the training data.

A standard assumption in BED is that the underlying model used to generate the data is *correct*. This means we simulate data from this model, which is assumed to perfectly represent the system we aim to learn about. The simulation process involves sampling parameters θ and possible outcomes y_t under the experimental designs ξ_t for $t = 1, \dots, T$.

Throughout this thesis, we train the policies in an online manner, meaning that data is simulated as the policy is updated. Future research can look into offline policy training, i.e. using pre-simulated design-outcome trajectories (e.g. from a random policy).

How sensitive are the results to the model? How can we reduce the sensitivity?

Since BED is a model-based approach, the results are inherently sensitive to (the accuracy of) the assumed model. The exact level of sensitivity will depend on the discrepancy between the assumed model and the real-world process, or how much we perturb the model.

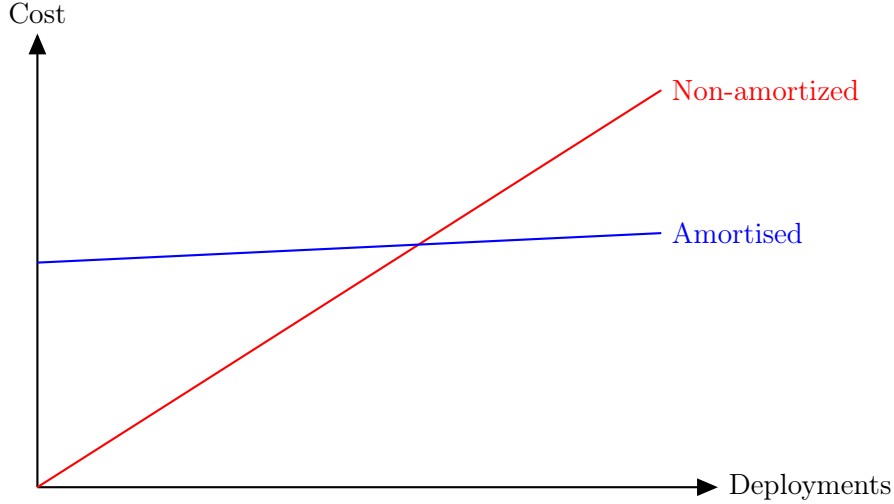


Figure 6.1: **Amortization over time.** The red line represents the cost of non-amortized (traditional) experimental design, which increases linearly with each experiment instance (deployment). The blue curve represents the cost of performing amortized BED: the initial expenditure is higher, but the cost per deployment is very small compared to the non-amortized approaches.

If the model is *misspecified*, the learned policy may perform poorly in real-world scenarios. To reduce sensitivity, we can use more flexible model families, incorporate robustness techniques (e.g., model averaging or adversarial training), or perform sensitivity analysis to understand how changes in the model affect performance. For further discussion, see Section 6.2.1.

What is the computational cost of the amortization? If that is taken into account, are policy-based BED approaches (like DAD) still faster? Under what conditions? Amortization incurs an upfront computational cost during policy training. Once trained, however, policy-based approaches like DAD can quickly generate designs without requiring any significant computation during the live experiment itself. This represents a trade-off that is illustrated in Figure 6.1: we shift the computational burden to the beginning, avoiding costly computations during the experiment.

Overall, the cost-efficiency of PB-BED depend on the specific use-case. The conditions favoring amortized approaches include: a) Large number of expected deployments; b) Availability of computational resources for upfront training; c) Scenarios where rapid deployment is crucial. If total computational cost is the primary concern, amortized methods become more efficient only as the number of deployments increases beyond the break-even point.

Why does the traditional myopic baseline perform so poorly? Table 1 in Chapter 2 shows that the myopic adaptive design strategy using variational posteriors performs on par with the static design strategy. This result is rather unexpected, as we would expect that an

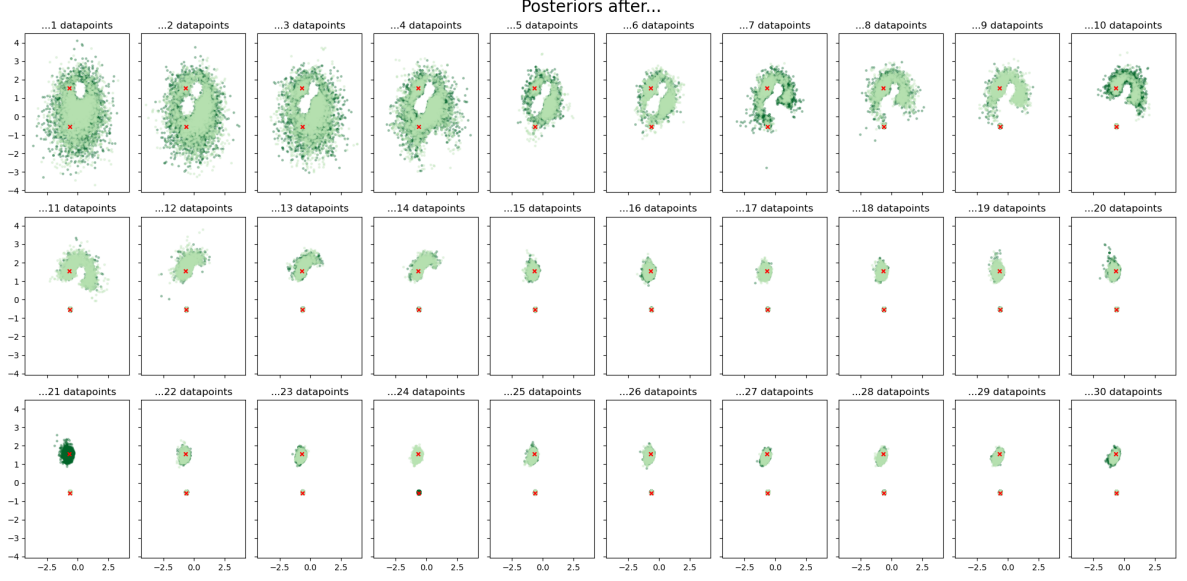


Figure 6.2: **Location finding experiment.** Example set of posteriors for $T = 30$ randomly sampled designs. We use Hamiltonian Monte Carlo (HMC), as implemented in Stan (Carpenter et al., 2017), to approximate the posteriors after each experiment iteration. The figure clearly shows that the posteriors are not Gaussian at the early stages of the experiment, when data is limited, meaning that standard mean field VI would perform poorly in this setting.

adaptive strategy, albeit myopic, would perform better than a static one. The reason for this under-performance was the fact that we used a simple Gaussian variational family (mean field variational inference), which is very far from the true posteriors in this model. Figure 6.2 shows an example of how these true posteriors look like. Additionally, one thing that was not discussed in the paper is that the posteriors are symmetric, which aligns with the model definition. Say we have $K = 2$ sources, and suppose we sample $\theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2$.

$$p(\theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2 \mid y, \xi) = \frac{p(\theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2)p(y \mid \theta_1 = \hat{\theta}_1, \theta_2 = \hat{\theta}_2)}{p(y \mid \xi)} \quad (6.1)$$

$$= \frac{p(\theta_1 = \hat{\theta}_2, \theta_2 = \hat{\theta}_1)p(y \mid \theta_1 = \hat{\theta}_2, \theta_2 = \hat{\theta}_1)}{p(y \mid \xi)} \quad (6.2)$$

This symmetry implies that we cannot distinguish between the two sources. To address this, we could either modify the model to make the parameters identifiable or “symmetrize” the posteriors to better align with the ground truth.

Re-running the same experiment using HMC, as implemented in Stan (Carpenter et al., 2017), to approximate the posteriors after each experiment iteration gives the following results:

$$\text{Lower bound : } 10.906 \pm 0.069 \quad \text{Upper bound } 41.078 \pm 3.182, \quad (6.3)$$

where we have used $L = 100\text{K}$ inner samples, and standard errors have been computed over 256 parameter rollouts. The code for these results is made publicly available on GitHub.

6.2 Limitations and future work

Recent advancements in Bayesian experimental design, specifically policy-based BED, have substantially improved the scalability, practicality, and versatility of the framework, making it applicable in real-time situations. The introduction of policy-based BED approaches (§2, §3, §4) represents a significant breakthrough in the field, opening up new avenues for further research in both methodology and applications. In this section, we explore some of the limitations in the current set-up and suggest potential areas for future research.

6.2.1 Model misspecification

BED is fundamentally a model-based framework, where domain experts and scientists typically construct the generative model $p(\theta)p(y|\theta, \xi)$. Many such models have been developed and successfully applied across various fields, including physics (Dushenko et al., 2020; Lored, 2004), bio-informatics (Vanlier et al., 2012), neuroscience (Shababo et al., 2013), epidemiology (Cook et al., 2008), and psychology (Vincent and Rainforth, 2017; Watson, 2017), among others. These applications typically involve mechanistic models, with BED playing a crucial role in system identification (Ljung, 2010).

A typical assumption in BED is that the underlying generative model is *correct*, i.e., it accurately represents the underlying data generating process (DGP). However, the models we work with are rarely a perfect representation of the DGP, leading to what is known as *model misspecification* or *model discrepancy*. Model misspecification can occur in two ways:

- **Prior misspecification:** If the prior $p(\theta)$ does not align with the observable data characteristics, it can lead to erroneous inferences and suboptimal designs. In BED, we often start without data, which means we cannot perform prior predictive checks to evaluate this alignment. Consequently, this can result in prior misspecification. For example, DasGupta and Studden (1991) and Ryan et al. (2016) show that prior misspecification leads to poor results even when EIG computation can be done in closed form. The impact is particularly pronounced in scenarios involving a batch of static experiments (as in Chapter §5) or in

fully amortised adaptive experiments (as in Chapters §2 and §3). However, in settings with repeated closed-loop experiments (as in Chapter §4), the repeated refinement of the posterior diminishes the influence of the initial prior, thus reducing concerns about how well it was specified (as long as the prior has full support).

- **Likelihood misspecification:** Likelihood misspecification occurs when the likelihood function, which models how the data is generated from the parameters, fails to accurately represent the true data-generating mechanism. Specifically, this means there is no parameter θ^* such that $p(y|\theta^*, \xi)$ matches the true distribution of the outcome $p(y|\xi)$ (Brynjarsdóttir and O’Hagan, 2014; Grünwald and Van Ommen, 2017; Kennedy and O’Hagan, 2001; Kleijn and Van der Vaart, 2012). This form of misspecification can seriously affect the estimated posterior distributions, leading to biased estimates and suboptimal experimental decisions. Unlike prior misspecification issues, which can be mitigated through repeated updates in closed-loop experiments, likelihood misspecification does not resolve with iterative updates, making it a more persistent challenge in the experimental design process.

While misspecification is inevitable in any Bayesian setting, it poses unique challenges in BED because the same (misspecified) model is used to both fit the data and guide the acquisition of new data. Although model misspecification cannot be completely eliminated, various strategies can be employed to mitigate its effects. However, their application in BED has been somewhat limited so far. Examples include:

Full Bayesian workflow for BED. Developing and implementing a comprehensive and automated Bayesian workflow (Gelman et al., 2013, 2020) that includes prior elicitation, rigorous model checking, and validation processes can significantly mitigate the risks associated with misspecification.

Robust objective functions. Developing objective functions that are robust to misspecification can help minimise its impact. For example, Go and Isaac (2022); Ouyang et al. (2016) modify the EIG to reduce the impact of prior specification. Defining objectives that do not assume the model is correct apriori is an important area for future research.

Better models. Coming up with good, realistic models presents substantial challenges, as it is a complex and often time-intensive process that requires translating sophisticated domain knowledge into precise mathematical formulations. However, developing or learning more accurate

and realistic likelihood models can significantly improve their adaptability to the complexities of real-world phenomena. This can be achieved through various approaches:

- **Advanced modelling techniques:** Hierarchical, non-parametric, or a combinations of learnt and mechanistic models can adapt better to real-world complexities. However, this often involves a trade-off with analytic tractability, as more complex models may not admit analytical or easy to approximate solutions.
- **Data-driven methods:** Data-driven methods, such as Variational Autoencoders (VAEs) (Kingma and Welling, 2014), Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), normalising flows (Papamakarios et al., 2021), diffusions (Song and Ermon, 2019; Vahdat et al., 2021), or transformer-based models (Vaswani et al., 2017), present possible alternatives to mechanistic models. However, these learnt methods typically require large amounts of training data to be effective, and the Bayesian treatment of all parameters in such large-scale models often becomes impractical. Simplifications to a few stochastic parameters (e.g. stochastic final layer) have been explored and may present a viable path forward (Sharma et al., 2023).
- **Hybrid methods:** Exploring a balanced integration of traditional mechanistic models with modern data-driven approaches could provide an interesting area for research.

6.2.2 Evaluating BED methods

Evaluating different BED strategies is a non-trivial task. Traditional metrics from supervised learning, such as F-scores, RMSE, accuracy, etc., are not directly applicable as they do not align with the core objective of BED, which is to optimise the data acquisition process itself, rather than to predict or classify data that has already been collected. The commonly used evaluation metric for BED is itself the EIG of a design strategy (or an estimate of it).

Whilst the EIG aligns with the objective of BED (it is the same metric we optimise), relying solely on EIG as an evaluation metric has significant limitations, including:

- **Model dependence.** This is closely related to the previous section. The EIG calculations depend heavily on the specific model used, which can be problematic if the model does not accurately represent the underlying processes (as discussed in the previous two subsections). As a result, the utility of EIG-focused evaluation is rather limited.
- **Difficulty dealing with implicit models.** Reliably estimating EIG for implicit models

can be particularly difficult, as it is usually not possible to determine the tightness or looseness of the learnt bounds.

- **Applicability to downstream tasks.** It is possible that the data collected through BED strategies might be used for a variety of downstream tasks that involve training different models from the one used in data collection.

There are two key aspects that future research should look to develop: metrics and benchmarks.

Development of appropriate metrics. To comprehensively assess the efficacy of BED strategies, new metrics are needed. These metrics should evaluate various aspects, including the quality and diversity of the collected data, performance on downstream tasks, potential fairness considerations, suitability for intended applications, and cost-effectiveness in terms of computational and sample resources (such as FLOPs and number of simulated samples required).

Development of appropriate benchmark tasks. The lack of a comprehensive suite of benchmarks in the BED field complicates the evaluation of new methods. Without established benchmarks, researchers must repeatedly re-implement tasks and algorithms for each new study, which is not only wasteful and time-consuming but also hinders the rapid evaluation and advancement of new BED methods. Other fields have demonstrated the value of benchmarks, such as ImageNet for image classification (Russakovsky et al., 2015) and Kodak for image compression (Franzen, 1999) in supervised learning, as well as benchmarks in simulation-based inference (Lueckmann et al., 2021) and reinforcement learning Bellemare et al. (2013); Duan et al. (2016). Developing similar benchmarks for BED could greatly facilitate methodological comparisons and foster innovation in the field.

6.2.3 PB-BED, model-based Reinforcement Learning and Active Inference

As noted in Section §1.5.3, the PB-BED framework we introduced in this thesis shares conceptual similarities with model-based Reinforcement Learning (MBRL) (Doll et al., 2012; Moerland et al., 2023) and Active Inference (Friston, 2010; Friston et al., 2017; Smith et al., 2022). In MBRL, an agent learns a model of the environment and uses this model to plan and make decisions that maximise long-term reward. PB-BED can be viewed as a special case of reward-free MBRL, where the model is provided by domain experts, and the objective is to learn about the model parameters. Active Inference, a framework from neuroscience, also shares similarities

with PB-BED in terms of minimising variational free energy (VFE), which is equivalent to maximising mutual information between sensory and internal states.

Despite these similarities, PB-BED, MBRL, and Active Inference have distinct assumptions and implications. Future work could explore the relationships between these frameworks in more depth. For example, investigating how PB-BED could be formulated within the Active Inference setting or how techniques from MBRL could be adapted to improve the efficiency, scalability and robustness of PB-BED. Additionally, the incorporation of model uncertainty in MBRL and Active Inference could be informed by the principles of optimal experimental design employed in PB-BED. Exploring these connections could lead to the development of more sample-efficient and robust learning algorithms for a wide range of applications.

6.2.4 Engineering

The introduction of deep learning modules into BED has considerably enhanced its capabilities. However, it necessitates specialised software, such as automatic differentiation (Paszke et al., 2019), stochastic gradient optimization (Kingma and Ba, 2014; Loshchilov and Hutter, 2017), and model tracking (Zaharia et al., 2018), which are primarily available in Python. Additionally, specialised hardware (GPU servers) might also be required.

Although the research community typically provides code for their proposed methods, as demonstrated in the published papers included in this thesis, the lack of standardisation across different frameworks often complicates the integration, application, and comparative evaluation of these methods. This fragmentation can significantly hinder progress in the field by creating barriers to quick experimentation and reproducibility.

Moreover, many existing scientific models (simulators) are written in a variety of programming languages, such as FORTRAN and MATLAB. This poses additional challenges for applying modern BED techniques, including those discussed in this thesis, which require differentiating through the model. This is not feasible across different languages or even across frameworks.

Addressing these engineering challenges requires long-term vision, investment, and commitment. A few initial steps (by no means exhaustive) include:

- **Development and support of a usable open-source BED framework:** Establishing a user-friendly, open-source framework for BED would be beneficial. A solid starting point would be to create a comprehensive suite of benchmark tasks, as previously discussed.
- **Providing the right incentives:** *(This point goes beyond the BED field and applies*

to the academic ML community more broadly.) The lack of incentives for developing and supporting software in academia, particularly in the academic machine learning community where software is crucial yet frequently undervalued, presents a challenge. Enhancing software quality in research settings likely requires a shift in academic and funding policies. Promoting positions that combine research with software development or hiring research engineers into academic teams are potential strategies that could shift the focus towards better software practices.

6.2.5 Real-world applications

The methodologies and improvements in adaptive BED discussed throughout this thesis hold significant potential for a wide range of scientific and industrial applications. The non-myopic aspect of DAD and iDAD (Chapters §2 and §3), combined with the ability to run in real time, opens the door to applications where decisions need to be made quickly, such as interactive surveys or adaptive educational tools. Partially amortised methods like Step-DAD (Chapter §4) allow us to leverage any available time between experiments to refine and adapt design policies, enhancing the efficacy of the design strategy over time. Running adaptive experiments via personalisation, including contextual information upfront, can be very valuable in cases where we run experiments on large batches of subjects, for example, in personalised medicine or targeted marketing (Chapter §5).

Realising this potential in practice requires addressing several key challenges, many of which have already been discussed in this section. We summarise them here:

- **More robust methods.** Enhancing the robustness of BED, particularly with regards to model misspecification (as discussed in §6.2.1).
- **Comprehensive evaluation.** Reliable and trustworthy deployment of BED strategies requires comprehensive evaluation metrics (as discussed in §6.2.2).
- **Engineering.** In addition to the engineering challenges in academia discussed in §6.2.4, real-world applications introduce additional challenges such as deploying in operational environments, managing large datasets, maintaining and updating models, and ensuring data privacy and security.

The main factor that could drive advancements across all these facets is **interdisciplinary collaboration**. Working closely with domain experts from academic or industry labs is the

only way to bridge theoretical research and practical implementation, and ensure that BED solutions are both scientifically rigorous and viable in practice.

6.3 Conclusion

In this thesis, we presented a series of advancements in the field of Bayesian experimental design (BED), with a particular focus on scalable and adaptive design strategies. We introduced several novel methods that leverage deep learning techniques to amortise the cost of sequential experimental design, enabling rapid and effective design decisions in real time. Deep Adaptive Design (DAD, Chapter §2) and its extension to implicit models (iDAD Chapter §3), have been shown to outperform existing approaches in terms of both the quality of the designs produced and the computational efficiency of the design process. Furthermore, we introduced a semi-amortised approach, Step-DAD (Chapter §4), which allows for periodic refinement of the design policy during the experiment itself, based on the actual data collected. This enhances the adaptability and robustness of the design strategy. In the context of batch experimentation, we formalised the problem of contextual optimization through the lens of BED and introduced CO-BED (Chapter §5), a general, model-agnostic framework for designing large-scale contextual experiments using information-theoretic principles.

The increased flexibility and efficiency that these new approaches introduce have the potential to make adaptive BED more accessible and widely applicable. However, challenges remain, particularly surrounding model misspecification, robust evaluation metrics, and the engineering considerations involved in deploying machine learning systems. Future research could explore combining mechanistic and data-driven models, defining application-oriented benchmarks for BED, and collaborating with domain experts to translate methodological advances into real-world applications.

We hope that the novel BED techniques we proposed in this thesis present a step towards a new era of automated and efficient experimentation strategies that, through practical applications, can drive scientific progress across a wide range of fields.

Chapter references

- Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. 182
- Jenný Brynjarsdóttir and Anthony O’Hagan. Learning about physical parameters: The importance of model discrepancy. *Inverse problems*, 30(11):114007, 2014. 180
- Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76, 2017. 178
- Alex R Cook, Gavin J Gibson, and Christopher A Gilligan. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008. 179
- A DasGupta and WJ Studden. Robust bayesian experimental designs in normal linear models. *The Annals of Statistics*, 19(3):1244–1256, 1991. 179
- Bradley B Doll, Dylan A Simon, and Nathaniel D Daw. The ubiquity of model-based reinforcement learning. *Current opinion in neurobiology*, 22(6):1075–1081, 2012. 182
- Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pages 1329–1338. PMLR, 2016. 182
- Sergey Dushenko, Kapildeb Ambal, and Robert D McMichael. Sequential bayesian experiment design for optically detected magnetic resonance of nitrogen-vacancy centers. *Physical Review Applied*, 14(5):054036, 2020. 179
- Rich Franzen. Kodak lossless true color image suite. *source: <http://r0k.us/graphics/kodak>*, 4(2):9, 1999. 182
- Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010. 182
- Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active inference: a process theory. *Neural computation*, 29(1):1–49, 2017. 182
- Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013. 180

- Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian workflow. *arXiv preprint arXiv:2011.01808*, 2020. 180
- Jinwoo Go and Tobin Isaac. Robust expected information gain for optimal bayesian experimental design using ambiguity sets. *38th Conference on Uncertainty in Artificial Intelligence*, 2022. 180
- Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 181
- Peter Grünwald and Thijs Van Ommen. Inconsistency of bayesian inference for misspecified linear models, and a proposal for repairing it. 2017. 180
- Marc C Kennedy and Anthony O’Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001. 180
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 183
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *ICLR*, 2014. 181
- Bas JK Kleijn and Aad W Van der Vaart. The bernstein-von-mises theorem under misspecification. 2012. 180
- Lennart Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010. 179
- Thomas J Loredo. Bayesian adaptive exploration. In *AIP Conference Proceedings*, volume 707, pages 330–346. American Institute of Physics, 2004. 179
- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 183
- Jan-Matthis Lueckmann, Jan Boelts, David Greenberg, Pedro Goncalves, and Jakob Macke. Benchmarking simulation-based inference. In *International conference on artificial intelligence and statistics*, pages 343–351. PMLR, 2021. 182
- Thomas M Moerland, Joost Broekens, Aske Plaat, Catholijn M Jonker, et al. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1):1–118, 2023. 182

- Long Ouyang, Michael Henry Tessler, Daniel Ly, and Noah Goodman. Practical optimal experiment design with probabilistic programs. *arXiv preprint arXiv:1608.05046*, 2016. 180
- George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021. 181
- 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. 183
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 182
- 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. 179
- Ben Shababo, Brooks Paige, Ari Pakman, and Liam Paninski. Bayesian inference and online experimental design for mapping neural microcircuits. In *Advances in Neural Information Processing Systems*, pages 1304–1312, 2013. 179
- Mrinank Sharma, Sebastian Farquhar, Eric Nalisnick, and Tom Rainforth. Do bayesian neural networks need to be fully stochastic? In *International Conference on Artificial Intelligence and Statistics*, pages 7694–7722. PMLR, 2023. 181
- Ryan Smith, Karl J Friston, and Christopher J Whyte. A step-by-step tutorial on active inference and its application to empirical data. *Journal of mathematical psychology*, 107:102632, 2022. 182
- Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 181
- Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in neural information processing systems*, 34:11287–11302, 2021. 181

- Joep Vanlier, Christian A Tiemann, Peter AJ Hilbers, and Natal AW van Riel. A Bayesian approach to targeted experiment design. *Bioinformatics*, 28(8):1136–1142, 2012. 179
- 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. 181
- Benjamin T Vincent and Tom Rainforth. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *Retrieved from psyarxiv.com/yehjb*, 2017. 179
- Andrew B Watson. Quest+: A general multidimensional bayesian adaptive psychometric method. *Journal of Vision*, 17(3):10–10, 2017. 179
- Matei Zaharia, Andrew Chen, Aaron Davidson, Ali Ghodsi, Sue Ann Hong, Andy Konwinski, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, et al. Accelerating the machine learning lifecycle with mlflow. *IEEE Data Eng. Bull.*, 41(4):39–45, 2018. 183

Chapter 7

Bibliography

List of Publications

Panagiotis Annadani, Yashas an f Tigas, Desi R Ivanova, Andrew Jesson, Yarin Gal, Adam Foster, and Stefan Bauer. Differentiable Multi-Target Causal Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 34263–34279. PMLR, 2023.

Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing Sequential Bayesian Experimental Design. In *International conference on machine learning*, pages 3384–3395. PMLR, 2021.

Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael U Gutmann, and Thomas Rainforth. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. *Advances in Neural Information Processing Systems*, 34, 2021.

Desi R Ivanova, Joel Jennings, Tom Rainforth, Cheng Zhang, and Adam Foster. CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 14445–14464. PMLR, 2023.

Desi R Ivanova, Marcel Hedman, Cong Guan, and Tom Rainforth. Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design. *ICLR 2024 Workshop on Data-centric Machine Learning Research (DMLR)*; Under review, 2024.

Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern Bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.

Marvin Schmitt, Desi R Ivanova, Daniel Habermann, Ullrich Köthe, Bürkner Paul-Christian, and Stefan T Radev. Leveraging Self-Consistency for Data-Efficient Amortized Bayesian Inference. In *International Conference on Machine Learning*. PMLR, 2024.

Combined Bibliography

- Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. {TensorFlow}: a system for {Large-Scale} machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pages 265–283, 2016.
- Lorenz Adlung, Yotam Cohen, Uria Mor, and Eran Elinav. Machine learning in clinical decision making. *Med*, 2(6):642–665, 2021.
- Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An optimistic perspective on offline reinforcement learning. In *International Conference on Machine Learning*, pages 104–114. PMLR, 2020.
- Ahsan S Alvi, Binxin Ru, Jan Calliess, Stephen J Roberts, and Michael A Osborne. Asynchronous batch bayesian optimisation with improved local penalisation. *arXiv preprint arXiv:1901.10452*, 2019.
- Panagiotis Annadani, Yashas anf Tigas, Desi R Ivanova, Andrew Jesson, Yarin Gal, Adam Foster, and Stefan Bauer. Differentiable Multi-Target Causal Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 34263–34279. PMLR, 2023.
- Theo Araujo, Natali Helberger, Sanne Kruikemeier, and Claes H De Vreese. In ai we trust? perceptions about automated decision-making by artificial intelligence. *AI & society*, 35: 611–623, 2020.
- Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Guided image generation with conditional invertible neural networks. *arXiv preprint arXiv:1907.02392*, 2019.
- Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.
- Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.
- AC Atkinson and VV Fedorov. The design of experiments for discriminating between two rival models. *Biometrika*, 62(1):57–70, 1975.

- Ahmed Attia, Alen Alexanderian, and Arvind K Saibaba. Goal-oriented optimal design of experiments for large-scale bayesian linear inverse problems. *Inverse Problems*, 34(9):095009, 2018.
- David Barber and Felix Agakov. The IM algorithm: a variational approach to information maximization. *Advances in Neural Information Processing Systems*, 16:201–208, 2003.
- Solon Barocas and Andrew D Selbst. Big data’s disparate impact. *Calif. L. Rev.*, 104:671, 2016.
- Joakim Beck, Ben Mansour Dia, Luis FR Espath, Quan Long, and Raul Tempone. Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain. *Computer Methods in Applied Mechanics and Engineering*, 334:523–553, 2018.
- Ishmael Belghazi, Sai Rajeswar, Aristide Baratin, R Devon Hjelm, and Aaron Courville. MINE: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.
- Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.
- James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyperparameter optimization. *Advances in neural information processing systems*, 24, 2011.
- José M Bernardo and Adrian FM Smith. *Bayesian theory*, volume 405. John Wiley & Sons, 2009.
- Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017.
- Tom Blau, Edwin V Bonilla, Iadine Chades, and Amir Dezfouli. Optimizing sequential experimental design with deep reinforcement learning. In *International Conference on Machine Learning*, pages 2107–2128. PMLR, 2022.
- David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.
- Eric Bogert, Aaron Schecter, and Richard T Watson. Humans rely more on algorithms than social influence as a task becomes more difficult. *Scientific reports*, 11(1):8028, 2021.
- George EP Box. Choice of response surface design and alphabetic optimality. Technical report, Wisconsin University-Madison Mathematics Research Center, 1982.

- Joan Fisher Box. Ra fisher and the design of experiments, 1922–1926. *The American Statistician*, 34(1):1–7, 1980.
- Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.
- Jenný Brynjarsdóttir and Anthony O’Hagan. Learning about physical parameters: The importance of model discrepancy. *Inverse problems*, 30(11):114007, 2014.
- Lukas Budach, Moritz Feuerpfel, Nina Ihde, Andrea Nathansen, Nele Noack, Hendrik Patzlaff, Felix Naumann, and Hazar Harmouch. The effects of data quality on machine learning performance. *arXiv preprint arXiv:2207.14529*, 2022.
- Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76, 2017.
- Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.
- Youngseog Chung, Ian Char, Willie Neiswanger, Kirthevasan Kandasamy, Andrew Oakleigh Nelson, Mark D Boyer, Egemen Kolenen, and Jeff Schneider. Offline contextual bayesian optimization for nuclear fusion. *arXiv preprint arXiv:2001.01793*, 2020.
- David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.
- Alex R Cook, Gavin J Gibson, and Christopher A Gilligan. Optimal observation times in experimental epidemic processes. *Biometrics*, 64(3):860–868, 2008.
- Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806, 2017.
- Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 1999.
- A DasGupta and WJ Studden. Robust bayesian experimental designs in normal linear models. *The Annals of Statistics*, 19(3):1244–1256, 1991.

- Mahasen B. Dehideniya, Christopher C. Drovandi, and James M. McGree. Optimal bayesian design for discriminating between models with intractable likelihoods in epidemiology. *Computational Statistics & Data Analysis*, 124:277–297, 2018. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2018.03.004>.
- Persi Diaconis and Donald Ylvisaker. Conjugate priors for exponential families. *The Annals of statistics*, pages 269–281, 1979.
- Bradley B Doll, Dylan A Simon, and Nathaniel D Daw. The ubiquity of model-based reinforcement learning. *Current opinion in neurobiology*, 22(6):1075–1081, 2012.
- Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pages 1329–1338. PMLR, 2016.
- Sergey Dushenko, Kapildeb Ambal, and Robert D McMichael. Sequential bayesian experiment design for optically detected magnetic resonance of nitrogen-vacancy centers. *Physical Review Applied*, 14(5):054036, 2020.
- Gustav Elfving. Optimum allocation in linear regression theory. *The Annals of Mathematical Statistics*, pages 255–262, 1952.
- Ronald Aylmer Fisher. *Design of experiments*. 1935.
- 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.
- 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.
- Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing Sequential Bayesian Experimental Design. In *International conference on machine learning*, pages 3384–3395. PMLR, 2021.
- Adam Evan Foster. *Variational, Monte Carlo and policy-based approaches to Bayesian experimental design*. PhD thesis, University of Oxford, 2021.

- Rich Franzen. Kodak lossless true color image suite. *source: <http://r0k.us/graphics/kodak>*, 4(2):9, 1999.
- Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010.
- Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, Giovanni Pezzulo, et al. Active inference and learning. *Neuroscience & Biobehavioral Reviews*, 68:862–879, 2016.
- Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active inference: a process theory. *Neural computation*, 29(1):1–49, 2017.
- Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.
- Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian workflow. *arXiv preprint [arXiv:2011.01808](https://arxiv.org/abs/2011.01808)*, 2020.
- Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984.
- Samuel Gershman and Noah Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014.
- Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, Aviv Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.
- Jinwoo Go and Tobin Isaac. Robust expected information gain for optimal bayesian experimental design using ambiguity sets. *38th Conference on Uncertainty in Artificial Intelligence*, 2022.
- Javier González, Michael Osborne, and Neil Lawrence. Glasses: Relieving the myopia of bayesian optimisation. In *Artificial Intelligence and Statistics*, pages 790–799. PMLR, 2016.
- Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint [arXiv:1406.2661](https://arxiv.org/abs/1406.2661)*, 2014.
- Peter Grünwald and Thijs Van Ommen. Inconsistency of bayesian inference for misspecified linear models, and a proposal for repairing it. 2017.

- Arthur Guez, David Silver, and Peter Dayan. Efficient Bayes-adaptive reinforcement learning using sample-based search. In *Advances in neural information processing systems*, pages 1025–1033, 2012.
- Jeff Guo, Bojana Ranković, and Philippe Schwaller. Bayesian optimization for chemical reactions. *Chimia*, 77(1/2):31–38, 2023.
- M Hamada, HF Martz, CS Reese, and AG Wilson. Finding near-optimal bayesian experimental designs via genetic algorithms. *The American Statistician*, 55(3):175–181, 2001.
- Cong Han and Kathryn Chaloner. Bayesian experimental design for nonlinear mixed-effects models with application to hiv dynamics. *Biometrics*, 60(1):25–33, 2004.
- W Keith Hastings. *Monte Carlo sampling methods using Markov chains and their applications*. Oxford University Press, 1970.
- Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(Jun):1809–1837, 2012.
- José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in neural information processing systems*, pages 918–926, 2014.
- Sebastian Höfer, Kostas Bekris, Ankur Handa, Juan Camilo Gamboa, Melissa Mozifian, Florian Golemo, Chris Atkeson, Dieter Fox, Ken Goldberg, John Leonard, et al. Sim2real in robotics and automation: Applications and challenges. *IEEE transactions on automation science and engineering*, 18(2):398–400, 2021.
- Zhang-Wei Hong, Tzu-Yun Shann, Shih-Yang Su, Yi-Hsiang Chang, Tsu-Jui Fu, and Chun-Yi Lee. Diversity-driven exploration strategy for deep reinforcement learning. *Advances in neural information processing systems*, 31, 2018.
- Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.
- Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *arXiv preprint arXiv:1605.09674*, 2016.
- Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013.

- Xun Huan and Youssef M Marzouk. Sequential bayesian optimal experimental design via approximate dynamic programming. *arXiv preprint arXiv:1604.08320*, 2016.
- Ben Hutchinson, Andrew Smart, Alex Hanna, Emily Denton, Christina Greer, Oddur Kjartansson, Parker Barnes, and Margaret Mitchell. Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 560–575, 2021.
- Frank Hutter, Holger Hoos, and Kevin Leyton-Brown. An evaluation of sequential model-based optimization for expensive blackbox functions. In *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, pages 1209–1216, 2013.
- Carl Hvarfner, Frank Hutter, and Luigi Nardi. Joint entropy search for maximally-informed bayesian optimization. *Advances in Neural Information Processing Systems*, 35:11494–11506, 2022.
- Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael U Gutmann, and Thomas Rainforth. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. *Advances in Neural Information Processing Systems*, 34, 2021.
- Desi R Ivanova, Joel Jennings, Tom Rainforth, Cheng Zhang, and Adam Foster. CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design. In *International Conference on Machine Learning*, pages 14445–14464. PMLR, 2023.
- Desi R Ivanova, Marcel Hedman, Cong Guan, and Tom Rainforth. Step-DAD: Semi-Amortized Policy-Based Bayesian Experimental Design. *ICLR 2024 Workshop on Data-centric Machine Learning Research (DMLR)*; Under review, 2024.
- Shali Jiang, Henry Chai, Javier Gonzalez, and Roman Garnett. Binoculars for efficient, nonmyopic sequential experimental design. In *International Conference on Machine Learning*, pages 4794–4803. PMLR, 2020.
- Marc C Kennedy and Anthony O’Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *ICLR*, 2014.

- Andreas Kirsch. Advancing deep active learning & data subset selection: Unifying principles with information-theory intuitions. *arXiv preprint arXiv:2401.04305*, 2024.
- Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32:7026–7037, 2019.
- Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal. Stochastic batch acquisition: A simple baseline for deep active learning. *arXiv preprint arXiv:2106.12059*, 2021.
- Bas JK Kleijn and Aad W Van der Vaart. The bernstein-von-mises theorem under misspecification. 2012.
- Steven Kleinegesse and Michael Gutmann. Efficient Bayesian experimental design for implicit models. *arXiv preprint arXiv:1810.09912*, 2018.
- 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. URL <http://proceedings.mlr.press/v119/kleinegesse20a.html>.
- Steven Kleinegesse and Michael U. Gutmann. Gradient-based bayesian experimental design for implicit models using mutual information lower bounds, 2021.
- Steven Kleinegesse, Christopher Drovandi, and Michael U Gutmann. Sequential bayesian experimental design for implicit models via mutual information. *arXiv preprint arXiv:2003.09379*, 2020.
- J Zico Kolter and Andrew Y Ng. Near-bayesian exploration in polynomial time. In *Proceedings of the 26th annual international conference on machine learning*, pages 513–520, 2009.
- Harold J Kushner. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *J of Basic Engineering*, 1964.
- Pawel Ladosz, Lilian Weng, Minwoo Kim, and Hyondong Oh. Exploration in deep reinforcement learning: A survey. *Information Fusion*, 85:1–22, 2022.
- Joel Lehman and Kenneth O Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2):189–223, 2011.

- Benjamin Letham and Eytan Bakshy. Bayesian optimization for policy search via online-offline experimentation. *Journal of Machine Learning Research*, 20(145):1–30, 2019.
- Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.
- Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956.
- Dennis V Lindley. *Bayesian statistics, a review*, volume 2. SIAM, 1972.
- Lennart Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010.
- Quan Long, Marco Scavino, Raúl Tempone, and Suojin Wang. Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations. *Computer Methods in Applied Mechanics and Engineering*, 259:24–39, 2013.
- Quan Long, Mohammad Motamed, and Raúl Tempone. Fast bayesian optimal experimental design for seismic source inversion. *Computer Methods in Applied Mechanics and Engineering*, 291:123–145, 2015.
- Thomas J Loredo. Bayesian adaptive exploration. In *AIP Conference Proceedings*, volume 707, pages 330–346. American Institute of Physics, 2004.
- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- Jan-Matthis Lueckmann, Jan Boelts, David Greenberg, Pedro Goncalves, and Jakob Macke. Benchmarking simulation-based inference. In *International conference on artificial intelligence and statistics*, pages 343–351. PMLR, 2021.
- David JC MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.
- David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.
- Tatsuya Matsushima, Hiroki Furuta, Yutaka Matsuo, Ofir Nachum, and Shixiang Gu. Deployment-efficient reinforcement learning via model-based offline optimization. *arXiv preprint arXiv:2006.03647*, 2020.

- Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021.
- Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.
- Petrus Mikkola, Osvaldo A Martin, Suyog Chandramouli, Marcelo Hartmann, Oriol Abril Pla, Owen Thomas, Henri Pesonen, Jukka Corander, Aki Vehtari, Samuel Kaski, et al. Prior knowledge elicitation: The past, present, and future. *arXiv preprint arXiv:2112.01380*, 2112, 2021.
- Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of bayesian methods for seeking the extremum. *Towards global optimization*, 2(117-129):2, 1978.
- Thomas M Moerland, Joost Broekens, Aske Plaat, Catholijn M Jonker, et al. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1): 1–118, 2023.
- Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. *arXiv preprint arXiv:1509.08731*, 2015.
- Peter Müller, Don A Berry, Andrew P Grieve, and Michael Krams. A bayesian decision-theoretic dose-finding trial. *Decision analysis*, 3(4):197–207, 2006.
- Willie Neiswanger, Kirthevasan Kandasamy, Barnabas Poczos, Jeff Schneider, and Eric Xing. Probo: Versatile bayesian optimization using any probabilistic programming language. *arXiv preprint arXiv:1901.11515*, 2019.
- Oliver Nelles and Oliver Nelles. *Nonlinear dynamic system identification*. Springer, 2020.
- Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Rectified max-value entropy search for bayesian optimization. *arXiv preprint arXiv:2202.13597*, 2022.
- 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.
- Eirini Ntoutsi, Pavlos Fafalios, Ujwal Gadiraju, Vasileios Iosifidis, Wolfgang Nejdl, Maria-Esther Vidal, Salvatore Ruggieri, Franco Turini, Symeon Papadopoulos, Emmanouil Krasanakis,

- et al. Bias in data-driven artificial intelligence systems—an introductory survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3):e1356, 2020.
- Long Ouyang, Michael Henry Tessler, Daniel Ly, and Noah Goodman. Practical optimal experiment design with probabilistic programs. *arXiv preprint arXiv:1608.05046*, 2016.
- 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.
- Anthony O’Hagan. Expert knowledge elicitation: subjective but scientific. *The American Statistician*, 73(sup1):69–81, 2019.
- Andrei Paleyes, Raoul-Gabriel Urma, and Neil D Lawrence. Challenges in deploying machine learning: a survey of case studies. *ACM computing surveys*, 55(6):1–29, 2022.
- Costas Papadimitriou. Optimal sensor placement methodology for parametric identification of structural systems. *Journal of sound and vibration*, 278(4-5):923–947, 2004.
- George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.
- 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.
- Amandalynne Paullada, Inioluwa Deborah Raji, Emily M Bender, Emily Denton, and Alex Hanna. Data and its (dis) contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11), 2021.
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.
- Ben Poole, Sherjil Ozair, Aäron van den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180, 2019.

- 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.
- Maria Priestley, Fionntán O’donnell, and Elena Simperl. A survey of data quality requirements that matter in ml development pipelines. *ACM Journal of Data and Information Quality*, 15(2):1–39, 2023.
- Stefan T Radev, Ulf K Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Köthe. Bayesflow: Learning complex stochastic models with invertible neural networks. *IEEE transactions on neural networks and learning systems*, 33(4):1452–1466, 2020.
- Stefan T Radev, Marvin Schmitt, Valentin Pratz, Umberto Picchini, Ullrich Köthe, and Paul-Christian Bürkner. Jana: Jointly amortized neural approximation of complex bayesian models. In *Uncertainty in Artificial Intelligence*, pages 1695–1706. PMLR, 2023.
- Howard Raiffa and Robert Schlaifer. Applied statistical decision theory. 1961.
- 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.
- Tom Rainforth, Adam Foster, Desi R Ivanova, and Freddie Bickford Smith. Modern Bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.
- Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.
- Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.
- 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.
- Sebastian Risi, Sandy D Vanderbleek, Charles E Hughes, and Kenneth O Stanley. How novelty search escapes the deceptive trap of learning to learn. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 153–160, 2009.

- Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.
- Stephane Ross, Brahim Chaib-draa, and Joelle Pineau. Bayes-adaptive pomdps. In *NIPS*, pages 1225–1232, 2007.
- Binxin Ru, Michael A Osborne, Mark McLeod, and Diego Granziol. Fast information-theoretic bayesian optimisation. In *International Conference on Machine Learning*, pages 4384–4392. PMLR, 2018.
- Binxin Ru, Xingchen Wan, Xiaowen Dong, and Michael Osborne. Interpretable neural architecture search via bayesian optimisation with weisfeiler-lehman kernels. *arXiv preprint arXiv:2006.07556*, 2020.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.
- Elizabeth G. Ryan, Christopher C. Drovandi, M. Helen Thompson, and Anthony N. Pettitt. Towards bayesian experimental design for nonlinear models that require a large number of sampling times. *Computational Statistics & Data Analysis*, 70:45–60, 2014. ISSN 0167-9473. doi: <https://doi.org/10.1016/j.csda.2013.08.017>. URL <https://www.sciencedirect.com/science/article/pii/S0167947313003149>.
- 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.
- Kenneth J Ryan. Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *Journal of Computational and Graphical Statistics*, 12(3):585–603, 2003.
- Thomas P Ryan and JP Morgan. Modern experimental design. *Journal of Statistical Theory and Practice*, 1(3-4):501–506, 2007.
- Syusuke Sano, Tadashi Kadowaki, Koji Tsuda, and Susumu Kimura. Application of bayesian optimization for pharmaceutical product development. *Journal of Pharmaceutical Innovation*, 15:333–343, 2020.

- Marvin Schmitt, Desi R Ivanova, Daniel Habermann, Ullrich Köthe, Bürkner Paul-Christian, and Stefan T Radev. Leveraging Self-Consistency for Data-Efficient Amortized Bayesian Inference. In *International Conference on Machine Learning*. PMLR, 2024.
- Ben Shababo, Brooks Paige, Ari Pakman, and Liam Paninski. Bayesian inference and online experimental design for mapping neural microcircuits. In *Advances in Neural Information Processing Systems*, pages 1304–1312, 2013.
- Amar Shah, Andrew G Wilson, and Zoubin Ghahramani. Bayesian optimization using student-t processes. In *NIPS Workshop on Bayesian Optimization*, 2013.
- Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.
- Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- Mrinank Sharma, Sebastian Farquhar, Eric Nalisnick, and Tom Rainforth. Do bayesian neural networks need to be fully stochastic? In *International Conference on Artificial Intelligence and Statistics*, pages 7694–7722. PMLR, 2023.
- Satinder Singh, Andrew G Barto, and Nuttapon Chentanez. Intrinsically motivated reinforcement learning. Technical report, MASSACHUSETTS UNIV AMHERST DEPT OF COMPUTER SCIENCE, 2005.
- Kirstine Smith. On the standard deviations of adjusted and interpolated values of an observed polynomial function and its constants and the guidance they give towards a proper choice of the distribution of observations. *Biometrika*, 12(1/2):1–85, 1918.
- Ryan Smith, Karl J Friston, and Christopher J Whyte. A step-by-step tutorial on active inference and its application to empirical data. *Journal of mathematical psychology*, 107:102632, 2022.
- Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.
- Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

- Susanne Still and Doina Precup. An information-theoretic approach to curiosity-driven reinforcement learning. *Theory in Biosciences*, 131:139–148, 2012.
- Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, pages 216–224. Elsevier, 1990.
- Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.
- Panagiotis Tigas, Yashas Annadani, Andrew Jesson, Bernhard Schölkopf, Yarin Gal, and Stefan Bauer. Interventions, where and how? experimental design for causal models at scale. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- Christian Toth, Lars Lorch, Christian Knoll, Andreas Krause, Franz Pernkopf, Robert Peharz, and Julius Von Kügelgen. Active bayesian causal inference. *Advances in Neural Information Processing Systems*, 35:16261–16275, 2022.
- Ryan Turner, David Eriksson, Michael McCourt, Juha Kiili, Eero Laaksonen, Zhen Xu, and Isabelle Guyon. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In *NeurIPS 2020 Competition and Demonstration Track*, pages 3–26. PMLR, 2021.
- Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in neural information processing systems*, 34:11287–11302, 2021.
- Jojanneke van den Berg, Andrew Curtis, and Jeannot Trampert. Optimal nonlinear bayesian experimental design: an application to amplitude versus offset experiments. *Geophysical Journal International*, 155(2):411–421, 2003.
- Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Joep Vanlier, Christian A Tiemann, Peter AJ Hilbers, and Natal AW van Riel. A Bayesian approach to targeted experiment design. *Bioinformatics*, 28(8):1136–1142, 2012.
- 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.

- A Helen Victoria and Ganesh Maragatham. Automatic tuning of hyperparameters using bayesian optimization. *Evolving Systems*, 12(1):217–223, 2021.
- Benjamin T Vincent and Tom Rainforth. The DARC toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *Retrieved from psyarxiv.com/yehjb*, 2017.
- Julius von Kügelgen, Paul K Rubenstein, Bernhard Schölkopf, and Adrian Weller. Optimal experimental design via bayesian optimization: active causal structure learning for gaussian process networks. *arXiv preprint arXiv:1910.03962*, 2019.
- Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.
- Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient Bayesian optimization. In *International Conference on Machine Learning*, pages 3627–3635. PMLR, 2017.
- Andrew B Watson. Quest+: A general multidimensional bayesian adaptive psychometric method. *Journal of Vision*, 17(3):10–10, 2017.
- Christopher K Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.
- Christina Winkler, Daniel E Worrall, Emiel Hoogeboom, and Max Welling. Learning likelihoods with conditional normalizing flows. 2019.
- Robert L Winkler. The assessment of prior distributions in bayesian analysis. *Journal of the American Statistical association*, 62(319):776–800, 1967.
- Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17(1):26–40, 2019.
- Keyi Wu, Peng Chen, and Omar Ghattas. An efficient method for goal-oriented linear bayesian optimal experimental design: Application to optimal sensor placemen. *arXiv preprint arXiv:2102.06627*, 2021.
- Matei Zaharia, Andrew Chen, Aaron Davidson, Ali Ghodsi, Sue Ann Hong, Andy Konwinski, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, et al. Accelerating the machine learning lifecycle with mlflow. *IEEE Data Eng. Bull.*, 41(4):39–45, 2018.

- Andrea Zanette, Kefan Dong, Jonathan N Lee, and Emma Brunskill. Design of experiments for stochastic contextual linear bandits. *Advances in Neural Information Processing Systems*, 34: 22720–22731, 2021.
- Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022.
- 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.
- Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, pages 737–744. IEEE, 2020.
- Sue Zheng, Jason Pacheco, and John Fisher. A robust approach to sequential information theoretic planning. In *International Conference on Machine Learning*, pages 5941–5949, 2018.
- Shijie Zhong, Wanggang Shen, Tommie Catanach, and Xun Huan. Goal-oriented bayesian optimal experimental design for nonlinear models using markov chain monte carlo. *arXiv preprint arXiv:2403.18072*, 2024.