CO-BED: Information-Theoretic Contextual Optimization via Bayesian Experimental Design

Desi R. Ivanova [1 †] **Joel Jennings** [2] **Tom Rainforth** [1] **Cheng Zhang** [2] **Adam Foster** [2]

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' demographics, 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. [1]Department of Statistics, University of Oxford [2]Microsoft Research. Correspondence to: Desi R Ivanova <desi.ivanova@stats.ox.ac.uk>, Adam Foster <adam.e.foster@microsoft.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

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 $p(y \mid \psi, \mathbf{a})$. 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 $\mathbf{a}$:

$$\begin{aligned}\mathrm{EIG}(\mathbf{a}) &= \mathbb{E}_{p(y|\mathbf{a})}\big[H[p(\psi)] - H[p(\psi \mid \mathbf{a}, y)]\big] \\ &= \mathbb{E}_{p(\psi)p(y|\psi,\mathbf{a})}\left[\log \frac{p(y \mid \psi, \mathbf{a})}{p(y \mid \mathbf{a})}\right], \end{aligned} \tag{1}$$

where $p(y \mid \mathbf{a}) = \mathbb{E}_{p(\psi)}[p(y \mid \psi, \mathbf{a})]$ is the Bayesian marginal distribution of the outcomes and is typically intractable. The EIG is equivalent to $I(\psi; y \mid \mathbf{a})$—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, $\mathrm{EIG}(\mathbf{A})$. 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 $p(\psi \mid \mathcal{D})$. To make inferences about any other quantity of interest, say ζ, we compute its posterior predictive distribution, $p(\zeta \mid \mathcal{D}) = \mathbb{E}_{p(\psi|\mathcal{D})}[p(\zeta \mid \mathcal{D}, \psi)]$.

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

$$\begin{aligned}&\mathrm{EIG}(\mathbf{a}) \geq \mathcal{L}(\mathbf{a}, U; L) \coloneqq \\ &\mathbb{E}_{p(\psi_{0:L})p(y|\mathbf{a},\psi_0)}\left[\log \frac{\exp(U(y,\psi_0))}{\frac{1}{L+1}\sum_\ell \exp(U(y,\psi_\ell))}\right]\end{aligned} \tag{2}$$

where $\psi_0 \sim p(\psi)$ is a primary or 'positive' sample from the prior, $y \sim p(y \mid \mathbf{a}, \psi_0)$ 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 $U : \mathcal{Y} \times \Psi \to \mathbb{R}$ is arbitrary and commonly referred to as a *critic*. The bound becomes tight in the limit as $L \to \infty$ for the *optimal* critic $U^*(y, \psi) = p(y \mid \psi, \mathbf{a}) + c(y)$, where $c(y)$ can be any function that only depends on the outcome y.

If the likelihood $p(y \mid \psi, \mathbf{a})$ 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.

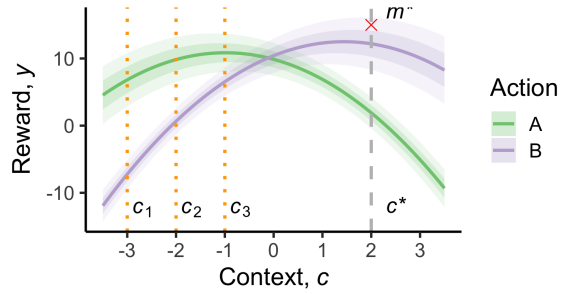


Figure 1. A stylized example. The dark (resp. light) shaded area shows our prior uncertainty about $y \mid \mathbf{a}, \mathbf{c}$ arising from uncertainty in ψ, measured by one (resp. two) st. dev. from the mean. We want to select actions in experimental contexts $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$ (orange dotted lines) whose outcomes will be most informative about $\mathbf{m}^*$ (red cross) in the evaluation context c^* (grey dashed line).

2.3. Max-value Entropy Search (MES)

The goal of (non-contextual) Bayesian optimization is to find the global maximizer $\mathbf{a}_\star = \arg\max_{\mathbf{a}\in\mathcal{A}} f(\mathbf{a})$ 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 $I(\mathbf{a}_\star; y \mid \mathbf{a})$—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 $m(\mathbf{c}) = \mathbb{E}[y \mid \mathbf{a}_\star, \mathbf{c}]$ the maximum value achievable in some context $\mathbf{c}$ where $\mathbf{a}_\star = \arg\max_{\mathbf{a}\in\mathcal{A}} \mathbb{E}[y \mid \mathbf{a}, \mathbf{c}]$ 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 $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_D)$, we seek to design a batch of actions $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_D)$ that will be maximally informative about the max rewards $\mathbf{m}^* = (m(\mathbf{c}_1^*), \ldots, m(\mathbf{c}_{D^*}^*))$ for a given set of *evaluation* contexts $\mathbf{C}^* = (\mathbf{c}_1^*, \ldots, \mathbf{c}_{D^*}^*)$ 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 thatthe 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 $\mathbf{C}$:

$$\begin{aligned}\text{CMV-EIG}(\mathbf{A}; \mathbf{C}, \mathbf{C}^*) &:= \mathbb{E}\left[\log \frac{p(\mathbf{y} \mid \mathbf{m}^*, \mathbf{C}, \mathbf{A})}{p(\mathbf{y} \mid \mathbf{C}, \mathbf{A})}\right] \\ &= I(\mathbf{m}^*; \mathbf{y} \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A}).\end{aligned} \tag{3}$$

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 $p(\psi)p(y, m^*|\psi, \mathbf{c}, \mathbf{c}^*, \mathbf{a})$, 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**

▷ Sample $\psi \sim p(\psi)$

▷ Sample $\mathbf{y}, \mathbf{m}^* \sim p(\mathbf{y}, \mathbf{m}^* \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A})$

▷ 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

▷ In parallel across $i = 1, \ldots, D$, for each experimental context $\mathbf{c}_i$, apply action $\mathbf{a}_i$ obtained in previous stage, receive outcome y_i. Set $\mathbf{y} = \{y_i\}_{i=1}^D$, $\mathcal{D} = \{\mathbf{y}, \mathbf{C}, \mathbf{A}\}$

▷ Estimate $p(\psi \mid \mathcal{D})$, use it to infer optimal actions $\mathbf{A}^* = \arg\max_{\mathbf{A}' \in \mathcal{A}^{D^*}} \mathbb{E}_{p(\psi|\mathcal{D})} \mathbb{E}\left[y \mid \mathbf{A}', \mathbf{C}^*, \psi\right]$

DEPLOYMENT PHASE

▷ Apply $\mathbf{A}^*$ to obtain outcomes $\mathbf{y}^*$

the joint distribution of outcomes and quantities of interest, marginalizing out the parameters: $p(\mathbf{y}, \mathbf{m}^* \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A}) = \mathbb{E}_{p(\psi)}\big[p(\mathbf{m}^* \mid \mathbf{C}^*, \psi) p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})\big]$. Similarly, the likelihood term is given by $p(\mathbf{y} \mid \mathbf{m}^*, \mathbf{C}, \mathbf{A}) = \mathbb{E}_{p(\psi|\mathbf{m}^*)}[p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})]$, and the marginal outcome in the denominator is $p(\mathbf{y} \mid \mathbf{C}, \mathbf{A}) = \mathbb{E}_{p(\psi)}\big[p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})\big]$.

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; Kleinegesse & Gutmann, 2021) to our CMV-EIG objective:

$$\mathcal{L}(\mathbf{A}, U; L) = \mathbb{E}\left[\log \frac{\exp(U(\mathbf{y}, \mathbf{m}_0^*))}{\frac{1}{L+1}\sum_\ell \exp(U(\mathbf{y}, \mathbf{m}_\ell^*))}\right] \tag{4}$$

$$\leq \text{CMV-EIG}(\mathbf{A}; \mathbf{C}, \mathbf{C}^*), \tag{5}$$

where the expectation is taken with respect to $p(\mathbf{y}, \mathbf{m}_0^* \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A}) p(\mathbf{m}_{1:L}^* \mid \mathbf{C}^*)$. The bound holds for any number of contrastive samples $L \geq 1$ and critic U, and becomes tight as $L \to \infty$ for the optimal one $U^*(\mathbf{y}, \mathbf{m}^*) = \log p(\mathbf{y} \mid \mathbf{m}^*, \mathbf{C}, \mathbf{A}) + c(\mathbf{y})$, 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 $\mathbf{y}, \mathbf{m}^* \sim p(\mathbf{y}, \mathbf{m}^* \mid \mathbf{C}, \mathbf{C}^*, \mathbf{A}) = \mathbb{E}_{p(\psi)}\big[p(\mathbf{m}^* \mid \psi, \mathbf{C}^*) p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})\big]$, and contrastive max-values $\mathbf{m}_{1:L}^* \sim \prod_{\ell=1}^L p(\mathbf{m}_\ell^* \mid \mathbf{C}^*)$, where $p(\mathbf{m}_\ell^* \mid \mathbf{C}^*) = \mathbb{E}_{p(\psi)}[p(\mathbf{m}_\ell^* \mid \psi, \mathbf{C}^*)]$. To obtain a sample from the joint, we first sample a parameter $\psi \sim p(\psi)$, then conditionally sample the outcomes $\mathbf{y} \sim p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})$ from our model.

To sample the corresponding max-rewards $\mathbf{m}^* \mid \psi \sim p(\mathbf{m}^* \mid \psi, \mathbf{C}^*)$, we distinguish three cases. First, in certain situations, we might be able to compute $\mathbf{A}_\star$ analytically, with many linear and parametric models falling into this category. For example, if $\mathbb{E}[y \mid \psi, \mathbf{a}, \mathbf{c}]$ depends linearly on $\mathbf{a}$, then computing m^* amounts to solving a linear program. Second, if $\mathcal{A}$ is discrete, we can determine the optimal actions, $\mathbf{A}_\star = \arg\max_{\mathbf{A}} \mathbb{E}\left[y \mid \psi, \mathbf{A}, \mathbf{C}^*\right]$ 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 $\mathbf{m}^* \mid \psi$ 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 $y \sim p(\mathbf{y} \mid \psi, \mathbf{C}, \mathbf{A})$ 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 $\pi_{\boldsymbol{\alpha}}$, where $\boldsymbol{\alpha} \in \mathbb{R}^{D \times K}$ 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:

$$\pi_{d,k} = \frac{\exp((\log \alpha_{d,k} + g_{d,k})/\tau)}{\sum_{j=1}^K \exp((\log \alpha_{d,j} + g_{d,j})/\tau)}, \tag{6}$$

where $g_{d,k} \sim \text{Gumbel}(0, 1)$ is Gumbel noise and $\tau > 0$ is a temperature hyper-parameter. We optimize the parameters $\boldsymbol{\alpha}$ and those of the critic network ϕ jointly with SGA by sampling $\mathbf{A} \sim \pi_\alpha$ and 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

As already discussed, CO-BED draws inspiration from several methods across somewhat separate fields to deliver a more general approach to contextual optimization. Our information objective (3) is most closely related to the MES approach to Bayesian optimization (Wang & Jegelka, 2017), but differs in two key ways: we are focused on CO, instead of finding a single maximizer, and we do use special properties of GP models for MI estimation. The InfoNCE lower bound (4), is rooted in the representation learning literature (van den Oord et al., 2018; Wu et al., 2018), and has also been used successful in standard 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

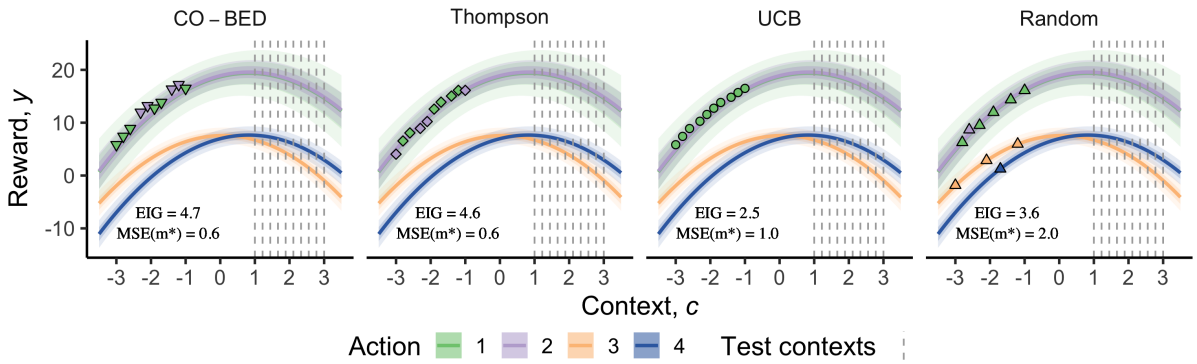


Figure 2. Discrete actions: designs and key metrics (further available in § A.2). The dashed grey lines show the evaluation contexts $\mathbf{C}^*$.

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; Houlsby 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 UCB_1 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_σ samples actions from $\mathcal{N}(0, \sigma)$, α in UCB_α is the confidence bound considered. Further details in § A.2.

Method	CMV-EIG ↑	MSE($\mathbf{m}^*$) ↓	MSE($\mathbf{A}$) ↓	Regret ↓
$\text{Random}_{0.2}$	5.407 ± 0.003	0.0041 ± 0.0001	0.544 ± 0.023	0.091 ± 0.002
$\text{Random}_{1.0}$	5.798 ± 0.004	0.0024 ± 0.0002	0.272 ± 0.018	0.060 ± 0.002
$\text{Random}_{2.0}$	4.960 ± 0.004	0.0042 ± 0.0002	0.450 ± 0.021	0.090 ± 0.002
UCB_0	5.774 ± 0.003	0.0069 ± 0.0005	0.747 ± 0.055	0.082 ± 0.002
UCB_1	5.876 ± 0.003	0.0030 ± 0.0002	0.338 ± 0.024	0.067 ± 0.002
UCB_2	5.780 ± 0.004	0.0031 ± 0.0002	0.378 ± 0.031	0.069 ± 0.002
Thompson	6.184 ± 0.004	0.0017 ± 0.0001	0.161 ± 0.007	0.051 ± 0.001
CO-BED	$\mathbf{6.527 \pm 0.003}$	$\mathbf{0.0014 \pm 0.0001}$	$\mathbf{0.143 \pm 0.018}$	$\mathbf{0.044 \pm 0.001}$

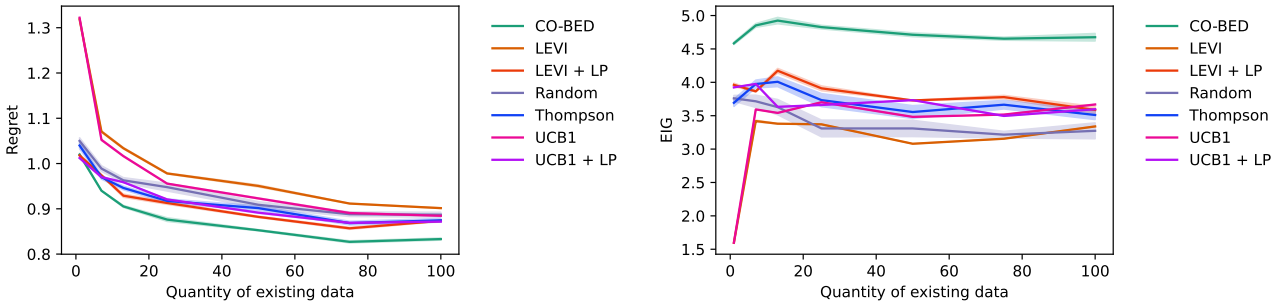


Figure 3. Results from the Gaussian Process example. Left: the regret evaluated on $\mathbf{C}^*$ after experimentation (↓ better). Right: the lower bound on the CMV-EIG at the end of training (↑ better). Plots show the mean ± 1 s.e. from 5 seeds. See Appendix A.3 for details.

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 $\psi : [-1, 1]^3 \to \mathbb{R}$, 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 $\psi = (\psi_1, \ldots, \psi_{20})$, is a 20-dimensional parameter vector, $a = 1, \ldots, 10$ 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-

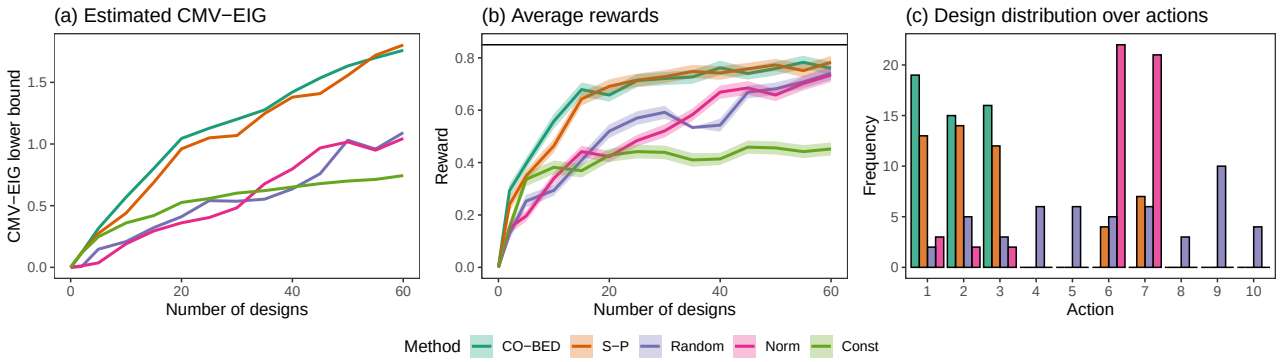


Figure 4. Linear contextual bandit results. CO-BED performs on par with the bespoke S-P algorithm both from (a) an information standpoint (↑ better) and (b) value of the rewards obtained during deployment (↑ better). The black horizontal line in (b) represents Supervised Learning (SL), an approximate 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 infeasability of Bayesian inference in this case. See Appendix A.5 for complete details. Our results in Figure 6

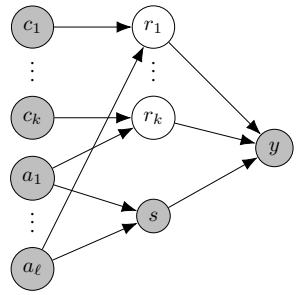


Figure 5. Causal graph considered in Section 5.4. The existence of edges from a_j to r_i is unknown. Unfilled nodes are not observed.

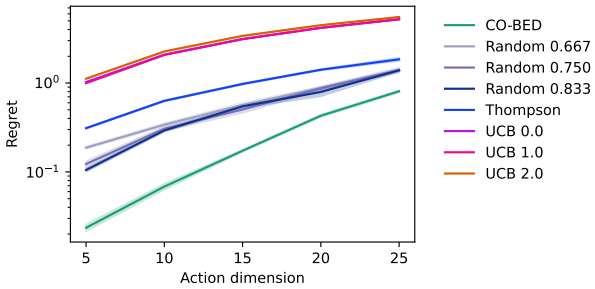


Figure 6. Results from the Unknown Causal Graph experiment showing regret on $\mathbf{C}^*$ after experimentation designed with different methods (↓ better). Plots show mean ±1 s.e. over 5 seeds.

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

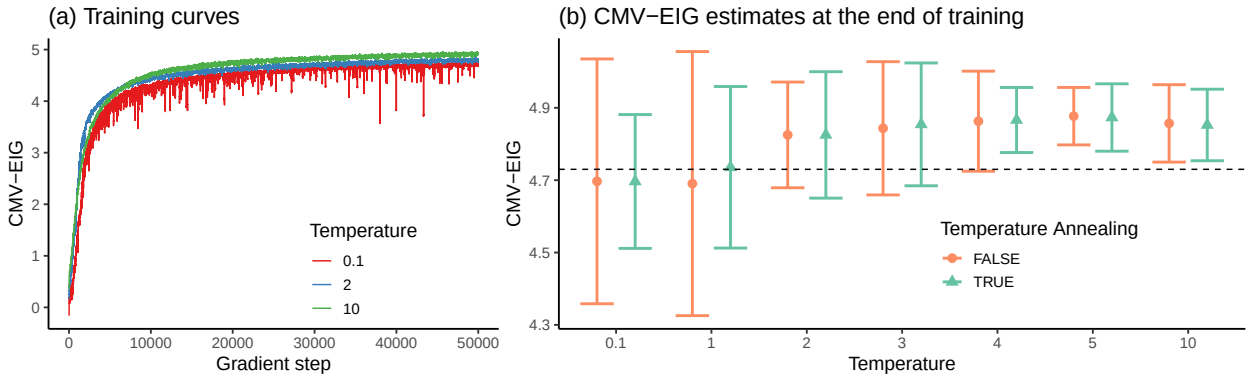


Figure 7. Robustness of the Gumbel-Softmax relaxation scheme. Panel (a) shows moving average estimates of CMV-EIG (4) (window size of 25) against gradient steps for different *fixed* values of τ (i.e. no annealing). Panel (b) presents CMV-EIG estimates computed at the end of training across 32 initializations 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

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: CambridgeUniversityPress*, 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-dimensryan2016reviewional 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

$$\mathbf{a}_{\text{post},i}^* = \arg\max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}_{\psi \sim p(\psi|,\mathbf{C},\mathbf{A},\mathbf{y})} \left[\mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi]\right] \tag{7}$$

$$m_{\text{post},i}^* = \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}_{\psi \sim p(\psi|,\mathbf{C},\mathbf{A},\mathbf{y})} \left[\mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi]\right]. \tag{8}$$

These can be compared with their counterparts under ψ_{true}

$$\mathbf{a}_{\text{true},i}^* = \arg\max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi_{\text{true}}] \tag{9}$$

$$m_{\text{true},i}^* = \max_{\mathbf{a}' \in \mathcal{A}} \mathbb{E}[y \mid \mathbf{a}', \mathbf{c}_i^*, \psi_{\text{true}}]; \tag{10}$$

giving us the 'treatment recovery' and 'reward recovery' MSEs, which are

$$L_{\text{treatment}} = \frac{1}{D^*} \sum_{i=1}^{D^*} \|\mathbf{a}_{\text{post},i}^* - \mathbf{a}_{\text{true},i}^*\|^2 \tag{11}$$

$$L_{\text{reward}} = \frac{1}{D^*} \sum_{i=1}^{D^*} |m_{\text{post},i}^* - m_{\text{true},i}^*|^2. \tag{12}$$

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

$$r_i = m_{\text{true},i}^* - \mathbb{E}[y \mid \mathbf{a}_{\text{post},i}^*, \mathbf{c}_i^*, \psi_{\text{true}}] \qquad r = \frac{1}{D^*} \sum_i r_i. \tag{13}$$

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 [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
$\text{UCB}_{0.0}$	1.735 ± 0.005	2.541 ± 0.104	0.513 ± 0.01	1.170 ± 0.036
$\text{UCB}_{1.0}$	2.514 ± 0.006	1.003 ± 0.043	0.496 ± 0.01	1.119 ± 0.035
$\text{UCB}_{2.0}$	2.504 ± 0.006	0.965 ± 0.045	0.497 ± 0.01	1.169 ± 0.037
Thompson	4.607 ± 0.007	0.620 ± 0.024	0.498 ± 0.01	1.112 ± 0.035
Random	3.573 ± 0.006	1.953 ± 0.070	0.503 ± 0.01	1.150 ± 0.036
Ours	$\mathbf{4.729 \pm 0.009}$	$\mathbf{0.594 \pm 0.025}$	0.501 ± 0.01	1.152 ± 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 $\mathbf{y}$. 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.

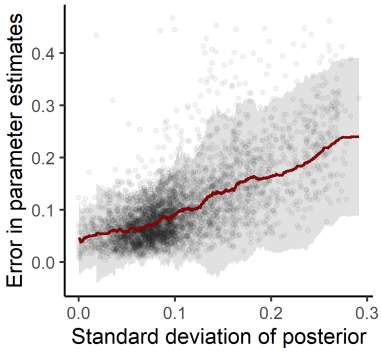


Figure 8. Posterior checks

Evaluation metrics details As discussed in the main text, we evaluate how well we can estimate $\mathbf{m}^*$ by sampling a ground truth $\tilde{\psi}$ 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 UCB_0 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):

$$\psi_1 \sim \mathcal{N}\left(\begin{pmatrix}5.00\\15.0\end{pmatrix}, \begin{pmatrix}9.00 & 0\\0 & 9.00\end{pmatrix}\right) \qquad \psi_2 \sim \mathcal{N}\left(\begin{pmatrix}5.00\\15.0\end{pmatrix}, \begin{pmatrix}2.25 & 0\\0 & 2.25\end{pmatrix}\right) \tag{14}$$

$$\psi_3 \sim \mathcal{N}\left(\begin{pmatrix}-2.0\\-1.0\end{pmatrix}, \begin{pmatrix}1.21 & 0\\0 & 1.21\end{pmatrix}\right) \qquad \psi_4 \sim \mathcal{N}\left(\begin{pmatrix}-7.0\\3.0\end{pmatrix}, \begin{pmatrix}1.21 & 0\\0 & 1.21\end{pmatrix}\right) \tag{15}$$

and reward (outcome) likelihoods:

$$y|\mathbf{c}, \mathbf{a}, \psi \sim \mathcal{N}\left(f(\mathbf{c}, \mathbf{a}, \psi), 0.1\right) \tag{16}$$

$$f(\mathbf{c}, \mathbf{a}, \psi) = -\mathbf{c}^2 + \beta(\mathbf{a}, \psi)\mathbf{c} + \gamma(\mathbf{a}, \psi) \tag{17}$$

$$\gamma = (\psi_{\mathbf{a},1} + \psi_{\mathbf{a},2} + 18)/2 \tag{18}$$

$$\beta = (\psi_{\mathbf{a},2} - \gamma + 9)/3 \tag{19}$$

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
$\text{UCB}_{0.0}$	1.740 ± 0.003	2.709 ± 0.058	0.500 ± 0.005	1.150 ± 0.017
$\text{UCB}_{1.0}$	2.508 ± 0.002	0.993 ± 0.016	0.498 ± 0.004	1.140 ± 0.007
$\text{UCB}_{2.0}$	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	$\mathbf{4.769 \pm 0.048}$	$\mathbf{0.628 \pm 0.025}$	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
$\text{Random}_{0.2}$	5.548 ± 0.044	0.0037 ± 0.0002	0.451 ± 0.033	0.083 ± 0.004
$\text{Random}_{1.0}$	5.654 ± 0.128	0.0031 ± 0.0004	0.343 ± 0.044	0.069 ± 0.006
$\text{Random}_{2.0}$	5.118 ± 0.163	0.0045 ± 0.0003	0.498 ± 0.032	0.086 ± 0.004
$\text{UCB}_{0.0}$	5.768 ± 0.002	0.0066 ± 0.0002	0.729 ± 0.022	0.082 ± 0.001
$\text{UCB}_{1.0}$	5.892 ± 0.006	0.0031 ± 0.0001	0.354 ± 0.013	0.068 ± 0.001
$\text{UCB}_{2.0}$	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	$\mathbf{6.538 \pm 0.008}$	$\mathbf{0.0013 \pm 0.0001}$	$\mathbf{0.131 \pm 0.006}$	$\mathbf{0.042 \pm 0.0001}$

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 contests, 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 UCB_1 and UCB_2 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:

$$\text{Prior:} \quad \psi = (\psi_0, \psi_1, \psi_2, \psi_3), \quad \psi_i \sim \text{Uniform}[0.1, 1.1] \text{ iid} \tag{20}$$

$$\text{Likelihood:} \quad y|\mathbf{c}, \mathbf{a}, \psi \sim \mathcal{N}(f(\psi, \mathbf{a}, \mathbf{c}), \sigma^2), \tag{21}$$

where

$$f(\psi, \mathbf{a}, \mathbf{c}) = \exp\left(-\frac{\big(a - g(\psi, \mathbf{c})\big)^2}{h(\psi, \mathbf{c})} - \lambda a^2\right) \quad g(\psi, \mathbf{c}) = \psi_0 + \psi_1 \mathbf{c} + \psi_2 \mathbf{c}^2 \quad h(\psi, c) = \psi_3. \tag{22}$$

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
$\text{Random}_{0.2}$	4.262 ± 0.004	0.0086 ± 0.0003	1.046 ± 0.041	0.120 ± 0.002
$\text{Random}_{1.0}$	4.264 ± 0.004	0.0068 ± 0.0003	0.799 ± 0.033	0.114 ± 0.002
$\text{Random}_{2.0}$	4.116 ± 0.003	0.0083 ± 0.0003	1.002 ± 0.044	0.127 ± 0.003
$\text{UCB}_{0.0}$	5.093 ± 0.004	0.0074 ± 0.0004	0.800 ± 0.047	0.097 ± 0.002
$\text{UCB}_{1.0}$	5.040 ± 0.004	0.0072 ± 0.0004	0.764 ± 0.041	0.097 ± 0.002
$\text{UCB}_{2.0}$	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	$\mathbf{5.642 \pm 0.003}$	$\mathbf{0.0034 \pm 0.0002}$	$\mathbf{0.065 \pm 0.002}$	$\mathbf{0.034 \pm 0.027}$

Table 6. Continuous treatment example: 60D design to learn about 59 evaluation contexts.

Method	EIG estimate	MSE($\mathbf{m}^*$)	MSE($\mathbf{A}$)	Regret
$\text{Random}_{0.2}$	6.033 ± 0.003	0.0026 ± 0.0001	0.307 ± 0.019	0.068 ± 0.002
$\text{Random}_{1.0}$	5.877 ± 0.004	0.0025 ± 0.0002	0.310 ± 0.023	0.064 ± 0.002
$\text{Random}_{2.0}$	6.153 ± 0.003	0.0022 ± 0.0002	0.226 ± 0.019	0.055 ± 0.002
$\text{UCB}_{0.0}$	6.106 ± 0.003	0.0056 ± 0.0004	0.586 ± 0.045	0.077 ± 0.002
$\text{UCB}_{1.0}$	6.200 ± 0.003	0.0027 ± 0.0003	0.305 ± 0.028	0.063 ± 0.002
$\text{UCB}_{2.0}$	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	$\mathbf{6.932 \pm 0.003}$	$\mathbf{0.0007 \pm 0.0001}$	$\mathbf{0.069 \pm 0.009}$	$\mathbf{0.033 \pm 0.001}$

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 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 $\text{Random}_0.2$, Random_1 and Random_2, 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 $c_i \overset{\text{i.i.d.}}{\sim} \text{Unif}(-1, 1)$. 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 $(\pm 0.8, \pm 0.8)$.

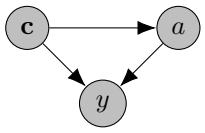
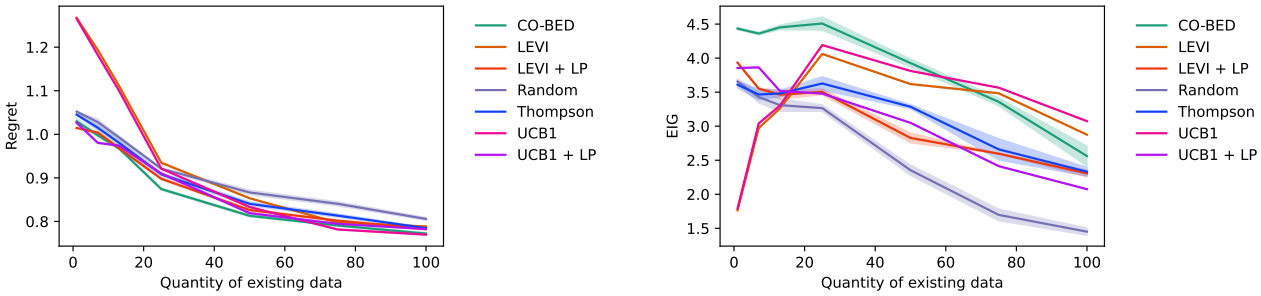


Figure 9. The form of the causal graph that generates observational data for Section 5.2.



(a) Regret

(b) EIG lower bound

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 $g_1, \ldots, g_G$ 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 $\Sigma_{a,c} = \text{diag}(10^{-9}, \ldots, 10^{-9}, 1, 10^{-9}, \ldots, 10^{-9})$, 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 $2, 5, 10, 15, \ldots, 60$.

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

$2\times$`input_dim`, $4\times$`input_dim`, `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 Bernoulli$(3/2k)$ and each component of $\theta \overset{\text{i.i.d.}}{\sim}$ 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 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

$$u_j := -1 + \sum_{i=1}^{k} c_i G_{ij}\theta_{ij}, \qquad a_j^* = \begin{cases} 1 \text{ if } u_j > 0, \\ 0 \text{ otherwise} \end{cases}, \qquad m^* = \sum_{j=1}^{\ell}\left(\sum_{i=1}^{k} c_i G_{ij}\theta_{ij}a_j^*\right) - a_j^*. \tag{23}$$

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.

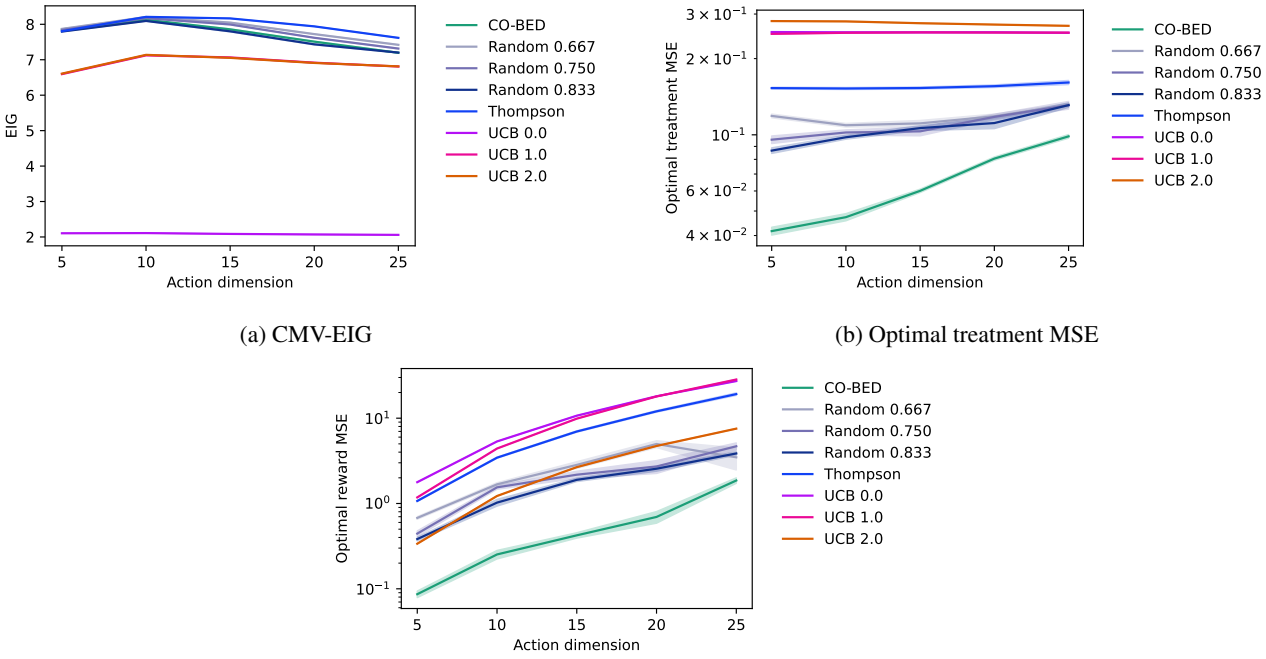
Parameter	Value
Intrinsic context dimension, k	8
Intrinsic treatment dimension, ℓ	$5, 10, 15, 20, 25$
Experimental batch size, D	200
Experimental context sampling probability	0.5
Evaluation batch size, D^*	255
Observation noise scale	0.25
Graph prior probability	$\frac{3}{16}$
Linear coefficient prior	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.