Active Bayesian Causal Inference

Christian Toth TU Graz
Lars Lorch ETH Zürich
Christian Knoll TU Graz
Andreas Krause ETH Zürich
Franz Pernkopf TU Graz

Robert Peharz* TU Graz
Julius von Kügelgen* MPI for Intelligent Systems, Tübingen
University of Cambridge

Abstract

Causal discovery and causal reasoning are classically treated as separate and consecutive tasks: one first infers the causal graph, and then uses it to estimate causal effects of interventions. However, such a two-stage approach is uneconomical, especially in terms of actively collected interventional data, since the causal query of interest may not require a fully-specified causal model. From a Bayesian perspective, it is also unnatural, since a causal query (e.g., the causal graph or some causal effect) can be viewed as a latent quantity subject to posterior inference—other unobserved quantities that are not of direct interest (e.g., the full causal model) ought to be marginalized out in this process and contribute to our epistemic uncertainty. In this work, we propose Active Bayesian Causal Inference (ABCI), a *fully-Bayesian active learning framework for integrated causal discovery and reasoning*, which jointly infers a posterior over causal models and queries of interest. In our approach to ABCI, we focus on the class of causally-sufficient, nonlinear additive noise models, which we model using Gaussian processes. We sequentially design experiments that are maximally informative about our target causal query, collect the corresponding interventional data, and update our beliefs to choose the next experiment. Through simulations, we demonstrate that our approach is more data-efficient than several baselines that only focus on learning the full causal graph. This allows us to accurately learn downstream causal queries from fewer samples while providing well-calibrated uncertainty estimates for the quantities of interest.

1 Introduction

Causal reasoning, that is, answering causal queries such as the effect of a particular intervention, is a fundamental scientific quest [3, 36, 39, 49]. A rigorous treatment of this quest requires a reference causal model, typically consisting at least of (i) a causal diagram, or directed acyclic graph (DAG), capturing the qualitative causal structure between a system's variables [55] and (ii) a joint distribution that is Markovian w.r.t. this causal graph [74]. Other frameworks additionally model (iii) the functional dependence of each variable on its causal parents in the graph [56, 82]. If the graph is not known from domain expertise, causal discovery aims to infer it from data [48, 74]. However, given only passively-collected observational data and no assumptions on the data-generating process, causal discovery is limited to recovering the Markov equivalence class (MEC) of DAGs implying the conditional independences present in the data [74]. Additional assumptions like linearity can render the graph identifiable [37, 60, 70, 85] but are often hard to falsify, thus leading to risk of

*Shared last author.

Correspondence to: {christian.toth,robert.peharz}@tugraz.at, jvk@tue.mpg.de
Code available at: https://www.github.com/chritoth/active-bayesian-causal-inference

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

misspecification. These shortcomings motivate learning from experimental (interventional) data, which enables recovering the true causal structure [16, 17, 31]. Since obtaining interventional data is costly in practice, we study the active learning setting, in which we sequentially design and perform interventions that are most informative for the target causal query [1, 26, 31, 32, 50, 78].

Classically, causal discovery and reasoning are treated as separate, consecutive tasks that are studied by different communities. Prior work on experimental design has thus focused either purely on causal reasoning—that is, how to best design experimental studies if the causal graph is known?—or purely on causal discovery, whenever the graph is unknown [35, 60]. In the present work, we consider the more general setting in which we are interested in performing causal reasoning but do not have access to a reference causal model a priori. In this case, causal discovery can be seen as a means to an end rather than as the main objective. Focusing on actively learning the *full* causal model to enable subsequent causal reasoning can thus be disadvantageous for two reasons. First, wasting samples on learning the full causal graph is suboptimal if we are only interested in specific aspects of the causal model. Second, causal discovery from small amounts of data entails significant epistemic uncertainty—for example, incurred by low statistical test power or multiple highly-scoring DAGs—which is not taken into account when selecting a single reference causal model [2, 21].

In this work, we propose *Active Bayesian Causal Inference* (ABCI), a fully-Bayesian framework for integrated causal discovery and reasoning with experimental design. The basic approach is to put a Bayesian prior over the causal model class of choice, and to cast the learning problem as Bayesian inference over the model posterior. Given the unobserved causal model, we formalize causal reasoning by introducing the *target causal query*, a function of the causal model that specifies the set of causal quantities we are interested in. The model posterior together with the query function induce a *query posterior*, which represents the result of our Bayesian learning procedure. It can be used, e.g., in downstream decision tasks or to derive a MAP solution or suitable expectation. To learn the query posterior, we follow the Bayesian optimal experimental design approach [10, 42] and sequentially choose admissible interventions on the true causal model that are most informative about our target query w.r.t. our current beliefs. Given the observed data, we then update our beliefs by computing the posterior over causal models and queries and use them to design the next experiment.

Since inference in the general ABCI framework is computationally highly challenging, we instantiate our approach for the class of causally-sufficient, nonlinear additive Gaussian noise models [37], which we model using Gaussian processes (GPs) [22, 81]. To perform efficient posterior inference in the combinatorial space of causal graphs, we use a recently proposed framework for differentiable Bayesian structure learning (DiBS) [45] that employs a continuous latent probabilistic graph representation. To efficiently maximise the information gain in the experiment design loop, we rely on Bayesian optimisation [46, 47, 72]. Overall, we highlight the following contributions:

- We propose ABCI as a flexible Bayesian active learning framework for efficiently inferring arbitrary sets of causal queries, subsuming causal discovery and reasoning as special cases (§ 3).
- We provide a fully Bayesian treatment for the flexible class of nonlinear additive Gaussian noise models by leveraging GPs, continuous graph parametrisations, and Bayesian optimisation (§ 4).
- We demonstrate that our approach scales to relevant problem sizes and compares favourably to baselines in terms of efficiently learning the graph, full SCM, and interventional distributions (§ 5).

2 Related Work

Causal discovery and reasoning have been widely studied in machine learning and statistics [27, 35, 60, 80]. Given an already collected set of observations, there is a large body of literature on learning causal structure, both in the form of a point estimate [9, 30, 43, 59, 70, 74] and a Bayesian posterior [2, 4, 12, 14, 21, 33, 45]. Given a known causal graph, previous work studies how to estimate treatment effects or counterfactuals [56, 66, 68]. When interventional data is yet to be collected, existing work primarily focuses on the specific task of structure learning—without its downstream use. The concept of (Bayesian) active causal discovery was first considered in discrete [50, 78] or linear [11, 53] models with closed-form marginal likelihoods and later extended to nonlinear causal mechanisms [77, 79], multi-target interventions [76], and general models by using hypothesis testing [23] or heuristics [67]. Graph theoretic works give insights on the interventions required for partial or full identifiability [15–17, 31, 38, 40, 69, 83].

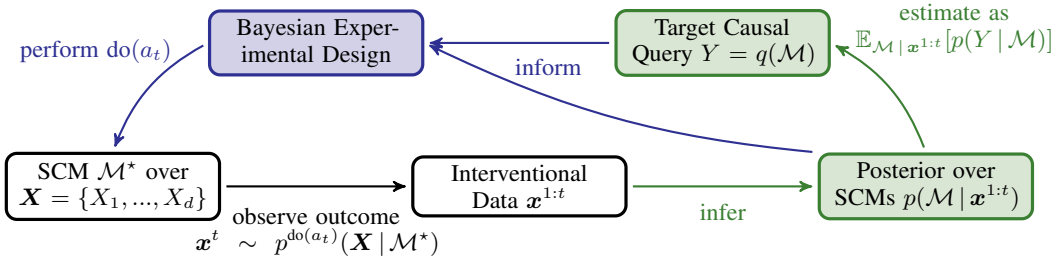

Figure 1: **Overview of the Active Bayesian Causal Inference (ABCI) framework.** At each time step t, we use Bayesian experimental design based on our current beliefs to choose a maximally informative intervention a_t to perform. We then collect a finite data sample from the interventional distribution induced by the environment, which we assume to be described by an unknown structural causal model (SCM) $\mathcal{M}^\star$ over a set of observable variables $\boldsymbol{X}$. Given the interventional data $\boldsymbol{x}^{1:t}$ collected from the true SCM $\mathcal{M}^\star$ and a prior distribution over the model class of consideration, we infer the posterior over a target causal query $Y = q(\mathcal{M})$ that can be expressed as a function of the causal model. For example, we may be interested in the graph (causal discovery), the presence of certain edges (partial causal discovery), the full SCM (causal model learning), a collection of interventional distributions or treatment effects (causal reasoning), or any combination thereof.

Beyond learning the complete causal graph, few prior works have studied active causal inference. Concurrent work of Tigas et al. [77] considers experimental design for learning a full SCM parameterised by neural networks. There are significant differences to our approach. In particular, our framework (§ 3) is not limited to the information gain over the full model and provides a fully Bayesian treatment of the functions and their epistemic uncertainty (§ 4). Agrawal et al. [1] consider actively learning a function of the causal graph under budget constraints, though not of the causal mechanisms and only for linear Gaussian models. Conversely, Rubenstein et al. [65] perform experimental design for learning the causal mechanisms after the causal graph has been inferred. Thus, while prior work considers causal discovery and reasoning as separate tasks, ABCI forms an integrated Bayesian approach for learning causal queries through interventions, reducing to previously studied settings in special cases. We further discuss related work in Appx. A.

3 Active Bayesian Causal Inference (ABCI) Framework

In this section, we first introduce the ABCI framework in generality and formalize its main concepts and distributional components, which are illustrated in Fig. 1. In § 4, we then describe our particular instantiation of ABCI for the class of causally sufficient nonlinear additive Gaussian noise models.

Notation. We use upper-case X and lower-case x to denote random variables and their realizations, respectively. Sets and vectors are written in bold face, $\boldsymbol{X}$ and $\boldsymbol{x}$. We use $p(\cdot)$ to denote different distributions, or densities, which are distinguished by their arguments.

Causal Model. To treat causality in a rigorous way, we first need to postulate a mathematically well-defined causal model. Historically hard questions about causality can then be reduced to epistemic questions, that is, what and how much is known about the causal model. A prominent type of causal model is the structural causal model (SCM) [56]. From a Bayesian perspective, an SCM can be viewed as a hierarchical data-generating process involving latent random variables.

Definition 1 (SCM)**.** An SCM $\mathcal{M}$ over observed endogenous variables $\boldsymbol{X} = \{X_1, \dots, X_d\}$ and unobserved exogenous variables $\boldsymbol{U} = \{U_1, \dots, U_d\}$ consists of structural equations, or mechanisms,

$$X_i := f_i(\mathbf{Pa}_i, U_i), \qquad \text{for} \quad i \in \{1, \dots, d\}, \tag{3.1}$$

which assign the value of each X_i as a deterministic function f_i of its direct causes, or causal parents, $\mathbf{Pa}_i \subseteq \boldsymbol{X} \setminus \{X_i\}$ and U_i; and a joint distribution $p(\mathbf{U})$ over the exogenous variables.

Associated with each SCM is a directed causal graph G with vertices $\boldsymbol{X}$ and edges $X_j \to X_i$ if and only if $X_j \in \mathbf{Pa}_i$, which we assume to be acyclic. Any acyclic SCM then induces a unique observational distribution $p(\boldsymbol{X} \mid \mathcal{M})$ over the endogenous variables $\boldsymbol{X}$, which is obtained as the pushforward measure of $p(\boldsymbol{U})$ through the causal mechanisms in Eq. (3.1).

Interventions. A crucial aspect of causal models such as SCMs is that they also model the effect of *interventions*—external manipulations to one or more of the causal mechanisms in Eq. (3.1)—which,

in general, are denoted using Pearl's do-operator [56] as $\text{do}(\{X_i = \tilde{f}_i(\mathbf{Pa}_i, U_i)\}_{i \in \mathcal{I}})$ with $\mathcal{I} \subseteq [d]$ and suitably chosen $\tilde{f}_i(\cdot)$. An intervention leads to a new SCM, the so-called interventional SCM, in which the relevant structural equations in Eq. (3.1) have been replaced by the new, manipulated ones. The interventional SCM thus induces a new distribution over the observed variables, the so-called interventional distribution, which is denoted by $p^{\text{do}(a)}(\boldsymbol{X} \mid \mathcal{M})$ with a denoting the (set of) intervention(s) $\{X_i = \tilde{f}_i(\mathbf{Pa}_i, U_i)\}_{i \in \mathcal{I}}$. Causal effects, that is, expressions like $\mathbb{E}[X_j | \text{do}(X_i = 3)]$, can then be derived from the corresponding interventional distribution via standard probabilistic inference.

Being Bayesian with Respect to Causal Models. The main epistemic challenge for causal reasoning stems from the fact that the true causal model $\mathcal{M}^\star$ is not or not completely known. The canonical response to such epistemic challenges is a Bayesian approach: place a prior $p(\mathcal{M})$ over causal models, collect data $\mathcal{D}$ from the true model $\mathcal{M}^\star$, and compute the posterior via Bayes rule:

$$p(\mathcal{M} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \mathcal{M})\, p(\mathcal{M})}{p(\mathcal{D})} = \frac{p(\mathcal{D} \mid \mathcal{M})\, p(\mathcal{M})}{\int p(\mathcal{D} \mid \mathcal{M})\, p(\mathcal{M})\, \mathrm{d}\mathcal{M}} \,. \tag{3.2}$$

A full Bayesian treatment over $\mathcal{M}$ is computationally delicate, to say the least. We require a way to parameterise the class of models $\mathcal{M}$ while being able to perform posterior inference over this model class. In this paper, we present a fully Bayesian approach for flexibly modelling nonlinear relationships (§ 4).

Bayesian Causal Inference. In the causal inference literature, the tasks of causal discovery and causal reasoning are typically considered separate problems. The former aims to learn (parts of) the causal model $\mathcal{M}^\star$, typically the causal graph $G^\star$, while the latter assumes that the relevant parts of $\mathcal{M}^\star$ are already known and aims to identify and estimate some query of interest, typically using only observational data. This separation suggests a two-stage approach of first performing causal discovery and then fixing the model for subsequent causal reasoning. From the perspective of uncertainty quantification and active learning, however, this distinction is unnatural because intermediate, unobserved quantities like the causal model do not contribute to the epistemic uncertainty in the final quantities of interest. Instead, we define a causal query function q, which specifies a *target causal query* $Y = q(\mathcal{M})$ as a function of the causal model $\mathcal{M}$. This view thus subsumes and generalises causal discovery and reasoning into a unified framework. For example, possible causal queries are:

Causal Discovery:	$Y = q_{\text{CD}}(\mathcal{M}) = G$, that is, learning the full causal graph G;
Partial Causal Discovery:	$Y = q_{\text{PCD}}(\mathcal{M}) = \phi(G)$, that is, learning some feature ϕ of the graph, such as the presence of a particular (set of) edge(s);
Causal Model Learning:	$Y = q_{\text{CML}}(\mathcal{M}) = \mathcal{M}$, that is, learning the full SCM $\mathcal{M}$;
Causal Reasoning:	$Y = q_{\text{CR}}(\mathcal{M}) = \{p^{\text{do}(\boldsymbol{X}_{\mathcal{I}(j)})}(X_j \mid \mathcal{M})\}_{j \in \mathcal{J}}$, that is, learning a set of interventional distributions induced by $\mathcal{M}$.[2]

Given a causal query, Bayesian inference naturally extends to our learning goal, the *query posterior*:

$$p(Y \mid \mathcal{D}) = \int p(Y \mid \mathcal{M})\, p(\mathcal{M} \mid \mathcal{D})\, \mathrm{d}\mathcal{M} = \mathbb{E}_{\mathcal{M} \mid \mathcal{D}}[\, p(Y \mid \mathcal{M})] \,, \tag{3.3}$$

where $p(Y \mid \mathcal{M})$ is a point mass at $q(\mathcal{M})$. Evidently, computing Eq. (3.3) constitutes a hard computational problem in general, as we need to marginalise out the causal model. In § 4, we introduce a practical implementation for a restricted causal model class, informed by this challenge.

Identifiability of causal models and queries. A crucial concept is that of *identifiability* of a model class, which refers to the ability to uniquely recover the true model in the limit of infinitely many observations from it [25].[3] In the context of our setting, if the class of causal models $\mathcal{M}$ is identifiable, the model posterior $p(\mathcal{M} \mid \mathcal{D})$ in Eq. (3.2) and hence also the query posterior $p(Y \mid \mathcal{D})$ in Eq. (3.3) will collapse and converge to a point mass on their respective true values $\mathcal{M}^\star$ and $q(\mathcal{M}^\star)$, given infinite data and provided the true model has non-zero mass under our prior, $p(\mathcal{M}^\star) > 0$. Given

[2] Here, the set $\mathcal{J}$ can be uncountable, subsuming interventional distributions for a continuous set of interventions, possibly on different variables. Thus, in this case the return value of q is a set of density functions. In practice, these are implicitly represented in the learned Bayesian models, see § 5.

[3] It is worth pointing out that the term "identifiability" is sometimes used differently in the causal inference literature: within causal discovery, it typically refers to *structure identifiability*, that is, recovering only the causal graph; in the context of causal reasoning, on the other hand, it typically refers to whether an interventional (or counterfactual) query can be *expressed in terms of known quantities*, usually involving only the observational distribution. Here, we will use the term in its (original) statistical sense to refer to *identifiability of models*.

only *observational* data, causal models are notoriously unidentifiable in general: without further assumptions on $p(\mathbf{U})$ and the structural form of Eq. (3.1), neither the graph nor the mechanisms can be recovered. In this case, $p(\mathcal{M} \mid \mathcal{D})$ may only converge to an equivalence class of models that cannot be further distinguished. Note, however, that even in this case, $p(Y \mid \mathcal{D})$ may still sometimes collapse, for example, if the Markov equivalence class (MEC) of graphs is identifiable (under causal sufficiency) and our query concerns the presence of a particular edge which is shared by all graphs in the MEC.

Active Learning with Sequential Interventions. Rather than collecting a large observational dataset, we seek to leverage experimental data, which can help resolve some of the aforementioned identifiability issues and facilitate learning our target causal query more quickly, even if the model is identifiable. Since obtaining experimental data is costly in practice, we study the active learning setting in which we sequentially design experiments in the form of interventions a_t.[4] At each time step t, the outcome of this experiment a_t is a batch $\boldsymbol{x}^t$ of N_t i.i.d. observations from the true interventional distribution:

$$\boldsymbol{x}^t = \{\boldsymbol{x}^{t,n}\}_{n=1}^{N_t}, \qquad \boldsymbol{x}^{t,n} \overset{\text{i.i.d.}}{\sim} p^{\mathrm{do}(a_t)}(\boldsymbol{X} \mid \mathcal{M}^\star) \tag{3.4}$$

Crucially, we design the experiment a_t to be *maximally informative* about our target causal query Y. In our Bayesian setting, this is naturally formulated as maximising the myopic information gain from the next intervention, that is, the mutual information between Y and the outcome $\boldsymbol{X}^t$ [10, 42]:

$$\max_{a_t} \mathrm{I}(Y; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) \tag{3.5}$$

where $\boldsymbol{X}^t$ follows the predictive interventional distribution of the Bayesian causal model ensemble at time $t-1$ under intervention a_t, which is given by

$$\boldsymbol{X}^t \sim p^{\mathrm{do}(a_t)}(\boldsymbol{X} \mid \boldsymbol{x}^{1:t-1}) \propto \int p^{\mathrm{do}(a_t)}(\boldsymbol{X} \mid \mathcal{M})\, p(\mathcal{M} \mid \boldsymbol{x}^{1:t-1})\, \mathrm{d}\mathcal{M}. \tag{3.6}$$

By maximising Eq. (3.5), we collect experimental data and infer our target causal query Y in a highly efficient, goal-directed manner.

4 Tractable ABCI for Nonlinear Additive Noise Models

Having described the general ABCI framework and its conceptual components, we now detail how to instantiate ABCI for a flexible model class that still allows for tractable, approximate inference. This requires us to specify (i) the class of causal models we consider in Eq. (3.1), (ii) the types of interventions a_t we consider at each step and the corresponding interventional likelihood in Eq. (3.4), (iii) our prior distribution $p(\mathcal{M})$ over models, (iv) how to perform tractable inference of the model posterior in Eq. (3.2), and finally (v) how to maximise the information gain in Eq. (3.5) for experimental design.

Model Class and Parametrisation. In the following, we consider nonlinear additive Gaussian noise models [37] of the form

$$X_i := f_i(\mathbf{Pa}_i) + U_i, \qquad \text{with} \qquad U_i \sim \mathcal{N}(0, \sigma_i^2) \qquad \text{for} \quad i \in \{1, \dots, d\}, \tag{4.1}$$

where the f_i's are smooth, nonlinear functions and the U_i's are assumed to be mutually independent. The latter corresponds to the assumption of causal sufficiency, or no hidden confounding. Any model $\mathcal{M}$ in this model class can be parametrised as a triple $\mathcal{M} = (G, \boldsymbol{f}, \boldsymbol{\sigma}^2)$, where G is a causal DAG, $\boldsymbol{f} = (f_1, \dots, f_d)$ is a vector of functions defined over the parent sets implied by G, and $\boldsymbol{\sigma}^2 = (\sigma_1^2, \dots, \sigma_d^2)$ contains the Gaussian noise variances. Provided that the f_i are nonlinear and not constant in any of their arguments, the model is identifiable almost surely [37, 61].

Interventional Likelihood. We support the realistic setting where only a subset $\boldsymbol{W} \subseteq \boldsymbol{X}$ of all variables are actionable, that is, can be intervened upon.[5] We consider hard interventions of the form $\mathrm{do}(a_t) = \mathrm{do}(\boldsymbol{X}_\mathcal{I} = \boldsymbol{x}_\mathcal{I})$ that fix a subset $\boldsymbol{X}_\mathcal{I} \subseteq \boldsymbol{W}$ to a constant $\boldsymbol{x}_\mathcal{I}$. Due to causal sufficiency, the interventional likelihood under such hard interventions a_t factorises over the causal graph G and is given by the g-formula [63] or truncated factorisation [74]:

$$p^{\mathrm{do}(a_t)}(\boldsymbol{X} \mid G, \boldsymbol{f}, \boldsymbol{\sigma}^2) = \mathbb{I}\{\boldsymbol{X}_\mathcal{I} = \boldsymbol{x}_\mathcal{I}\} \prod_{j \notin \mathcal{I}} p(X_j \mid f_j(\mathbf{Pa}_j^G), \sigma_j^2). \tag{4.2}$$

[4] Note that restricting to $a_t = \varnothing$ amounts to learning from observational data as a special case.

[5] In principle, the set of actionable variables might even change over time, in which case they are denoted $\boldsymbol{W}_t$.

The last term in Eq. (4.2) is given by $\mathcal{N}(X_j \mid f_j(\mathbf{Pa}_j^G), \sigma_j^2)$, due to the Gaussian noise assumption. Let $\boldsymbol{x}^{1:t}$ be the entire dataset, collected up to time t. The likelihood of $\boldsymbol{x}^{1:t}$ is then given by

$$p(\boldsymbol{x}^{1:t} \mid G, \boldsymbol{f}, \boldsymbol{\sigma}^2) = \prod_{\tau=1}^{t} p^{\mathrm{do}(a_\tau)}(\boldsymbol{x}^{\tau} \mid G, \boldsymbol{f}, \boldsymbol{\sigma}^2) = \prod_{\tau=1}^{t} \prod_{n=1}^{N_t} p^{\mathrm{do}(a_\tau)}(\boldsymbol{x}^{\tau,n} \mid G, \boldsymbol{f}, \boldsymbol{\sigma}^2). \tag{4.3}$$

Structured Model Prior. To specify our prior, we distinguish between root nodes X_i, for which $\mathbf{Pa}_i = \varnothing$ and thus $f_i = \text{const}$, and non-root nodes X_j. For a given causal graph G, we denote the index set of root nodes by $\mathbf{R}(G) = \{i \in [d] : \mathbf{Pa}_i^G = \varnothing\}$ and that of non-root nodes by $\mathbf{NR}(G) = [d] \setminus \mathbf{R}(G)$. We then place the following structured prior over SCMs $\mathcal{M} = (G, \boldsymbol{f}, \boldsymbol{\sigma}^2)$:

$$p(\mathcal{M}) = p(G)\, p(\boldsymbol{f}, \boldsymbol{\sigma}^2 \mid G) = p(G) \prod_{i \in \mathbf{R}(G)} p(f_i, \sigma_i^2 \mid G) \prod_{j \in \mathbf{NR}(G)} p(f_j \mid G) p(\sigma_j^2 \mid G)\,. \tag{4.4}$$

Here, $p(G)$ is a prior over graphs and $p(\boldsymbol{f}, \boldsymbol{\sigma}^2 \mid G)$ is a prior over the functions and noise variances. We factorise our prior conditional on G as in Eq. (4.4) not only to allow for a separate treatment of root vs. non-root nodes, but also to share priors across similar graphs. Whenever $\mathbf{Pa}_i^{G_1} = \mathbf{Pa}_i^{G_2}$, we set $p(f_i, \sigma_i^2 \mid G_1) = p(f_i, \sigma_i^2 \mid G_2)$. As a consequence, the posteriors are also shared, which substantially reduces the computational cost in practice (see Appx. E.2 for details). Our prior also encodes the beliefs that $\{f_i, \sigma_i^2\} \perp\!\!\!\perp \{f_{i'}, \sigma_{i'}^2\} \mid G$ for $i \neq i' \in [d]$ and that $f_j \perp\!\!\!\perp \sigma_j^2 \mid G$ for $j \in \mathbf{NR}(G)$ which is motivated by the principle of independent causal mechanisms [60] and the causal sufficiency assumption. Our specific choices for the different factors on the RHS of Eq. (4.4) are guided by ensuring tractable inference and described in more detail below.

Model Posterior. Given collected data $\boldsymbol{x}^{1:t}$, we can update our beliefs and quantify our uncertainty in $\mathcal{M}^\star$ by inferring the posterior $p(\mathcal{M} \mid \boldsymbol{x}^{1:t})$ over SCMs $\mathcal{M} = (G, \boldsymbol{f}, \boldsymbol{\sigma}^2)$, which can be written as[6]

$$p(\mathcal{M} \mid \boldsymbol{x}^{1:t}) = p(G \mid \boldsymbol{x}^{1:t}) \prod_{i \in \mathbf{R}(G)} p(f_i, \sigma_i^2 \mid \boldsymbol{x}^{1:t}, G) \prod_{j \in \mathbf{NR}(G)} p(f_j, \sigma_j^2 \mid \boldsymbol{x}^{1:t}, G)\,. \tag{4.5}$$

For root nodes $i \in \mathbf{R}(G)$, posterior inference given the graph is straightforward. We have $f_i = \text{const}$, so f_i can be viewed as the mean of U_i. We thus place conjugate normal-inverse-gamma N-$\Gamma^{-1}(\mu_i, \lambda_i, \alpha_i^{\mathrm{R}}, \beta_i^{\mathrm{R}})$ priors on $p(f_i, \sigma_i^2 \mid G)$, which allows us to analytically compute the root node posteriors $p(f_i, \sigma_i^2 \mid \boldsymbol{x}^{1:t}, G)$ in Eq. (4.5) given the hyperparameters $(\boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{\alpha}^{\mathrm{R}}, \boldsymbol{\beta}^{\mathrm{R}})$ [51].

The posteriors over graphs and non-root nodes $j \in \mathbf{NR}(G)$ are given by

$$p(G \mid \boldsymbol{x}^{1:t}) = \frac{p(\boldsymbol{x}^{1:t} \mid G)\, p(G)}{p(\boldsymbol{x}^{1:t})}\,, \qquad p(f_j, \sigma_j^2 \mid \boldsymbol{x}^{1:t}, G) = \frac{p(\boldsymbol{x}^{1:t} \mid G, f_j, \sigma_j^2)\, p(f_j, \sigma_j^2 \mid G)}{p(\boldsymbol{x}^{1:t} \mid G)}\,. \tag{4.6}$$

Computing these posteriors is more involved and discussed in the following.

4.1 Addressing Challenges for Posterior Inference with GPs and DiBS

The posterior distributions in Eq. (4.6) are intractable to compute in general due to the marginal likelihood and evidence terms $p(\boldsymbol{x}^{1:t} \mid G)$ and $p(\boldsymbol{x}^{1:t})$, respectively. In the following, we will address these challenges by means of appropriate prior choices and approximations.

Challenge 1: Marginalising out the Functions. The marginal likelihood $p(\boldsymbol{x}^{1:t} \mid G)$ reads

$$p(\boldsymbol{x}^{1:t} \mid G) = \int p(\boldsymbol{x}^{1:t} \mid G, f_j, \sigma_j^2)\, p(f_j \mid G)\, p(\sigma_j^2 \mid G)\, \mathrm{d}f_j\, \mathrm{d}\sigma_j^2 \tag{4.7}$$

and requires evaluating integrals over the function domain. We use Gaussian processes (GPs) [81] as an elegant way to solve this problem, as GPs flexibly model *nonlinear* functions while offering convenient analytical properties. Specifically, we place a $\mathcal{GP}(0, k_j^G(\cdot,\cdot))$ prior on $p(f_j|G)$, where $k_j^G(\cdot,\cdot)$ is a covariance function over the parents of X_j with kernel parameters $\boldsymbol{\kappa}_j$. As is common, we refer to $(\boldsymbol{\kappa}_j, \sigma_j^2)$ as the GP-hyperparameters. In addition, we place Gamma$(\alpha_j^\sigma, \beta_j^\sigma)$ and Gamma $(\boldsymbol{\alpha}_j^\kappa, \boldsymbol{\beta}_j^\kappa)$ priors on $p(\sigma_i^2 \mid G)$ and $p(\boldsymbol{\kappa}_i \mid G)$ and collect their parameters in $(\boldsymbol{\alpha}^{\mathrm{GP}}, \boldsymbol{\beta}^{\mathrm{GP}})$.

[6] To avoid further complicating the notation, we write all posteriors and likelihoods in terms of the full data $\boldsymbol{x}^{1:t}$. However, only observations of X_i and $X_j \mid \mathbf{Pa}_j^G$ matter for $i \in \mathbf{R}(G)$ and $j \in \mathbf{NR}(G)$.

The graphical model underlying all variables and hyperparameters is shown in Fig. 2. For our model class, GPs provide closed-form expressions for the GP-marginal likelihood $p(\boldsymbol{x}^{1:t} \mid G, \sigma_j^2, \boldsymbol{\kappa}_j)$, as well as for the GP posteriors $p(f_j \mid \boldsymbol{x}^{1:t}, G, \sigma_j^2, \boldsymbol{\kappa}_j)$ and the predictive posteriors over observations $p(\boldsymbol{X} \mid \boldsymbol{x}^{1:t}, G, \boldsymbol{\sigma}^2, \boldsymbol{\kappa})$ [81], see Appx. B for details.

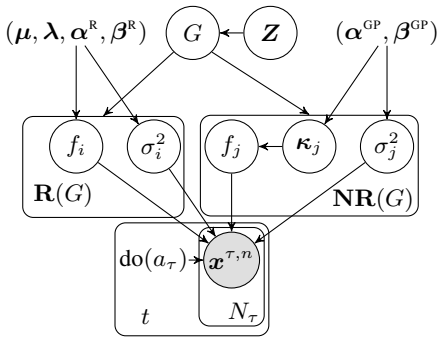

Figure 2: Graphical model of GP-DiBS-ABCI.

Challenge 2: Marginalising out the GP-Hyperparameters. While GPs allow for exact posterior inference conditional on a fixed instance of $(\sigma_j^2, \boldsymbol{\kappa}_j)$, evaluating expressions such as $p(f_j \mid \boldsymbol{x}^{1:t}, G)$ requires marginalising out these GP-hyperparameters from the GP-posterior. In general, this is intractable to do exactly, as there is no analytical expression for $p(\sigma_j^2, \boldsymbol{\kappa}_j \mid \boldsymbol{x}^{1:t}, G)$. To tackle this, we approximate such terms using a maximum a posteriori (MAP) point estimate $(\hat{\sigma}_j^2, \hat{\boldsymbol{\kappa}}_j)$ obtained by performing gradient ascent on the unnormalised log posterior

$$\nabla \log p(\sigma_j^2, \boldsymbol{\kappa}_j \mid \boldsymbol{x}^{1:t}, G) = \nabla \log p(\boldsymbol{x}^{1:t} \mid G, \sigma_j^2, \boldsymbol{\kappa}_j) + \nabla \log p(\sigma_j^2, \boldsymbol{\kappa}_j \mid G) \tag{4.8}$$

according to a predefined update schedule, see Alg. 1. More specifically,

$$p(f_j \mid \boldsymbol{x}^{1:t}, G) = \int p(f_j \mid \boldsymbol{x}^{1:t}, G, \sigma_j^2, \boldsymbol{\kappa}_j) p(\sigma_j^2, \kappa_j \mid \boldsymbol{x}^{1:t}, G) \, \mathrm{d}\sigma_j^2 \, \mathrm{d}\boldsymbol{\kappa}_j \approx p(f_j \mid \boldsymbol{x}^{1:t}, G, \hat{\sigma}_j^2, \hat{\boldsymbol{\kappa}}_j)$$

Challenge 3: Marginalising out the Causal Graph. The evidence $p(\boldsymbol{x}^{1:t})$ is given by

$$p(\boldsymbol{x}^{1:t}) = \sum_G p(\boldsymbol{x}^{1:t} \mid G) \, p(G) \tag{4.9}$$

and involves a summation over all possible DAGs G. This becomes intractable for $d \geq 5$ variables as the number of DAGs grows super-exponentially in the number of variables [64]. To address this challenge, we employ the recently proposed DiBS framework [45]. By introducing a continuous prior $p(\boldsymbol{Z})$ that models G via $p(G \mid \boldsymbol{Z})$ and simultaneously enforces acyclicity of G, Lorch et al. [45] show that we can efficiently infer the discrete posterior $p(G \mid \boldsymbol{x}^{1:t})$ via $p(\boldsymbol{Z} \mid \boldsymbol{x}^{1:t})$ as

$$\mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\phi(G)\right] = \mathbb{E}_{\boldsymbol{Z} \mid \boldsymbol{x}^{1:t}} \left[\frac{\mathbb{E}_{G \mid \boldsymbol{Z}} \left[p(\boldsymbol{x}^{1:t} \mid G) \, \phi(G) \right]}{\mathbb{E}_{G \mid \boldsymbol{Z}} \left[p(\boldsymbol{x}^{1:t} \mid G) \right]} \right] \tag{4.10}$$

where ϕ is some function of the graph. Since $p(\boldsymbol{Z} \mid \boldsymbol{x}^{1:t})$ is a continuous density with tractable gradient estimators, we can leverage efficient variational inference methods such as Stein Variational Gradient Descent (SVGD) for approximate inference [44]. Additional details on DiBS are given in Appx. D.

4.2 Approximate Bayesian Experimental Design with Bayesian Optimisation

Following § 3, our goal is to perform experiments a_t that are maximally informative about our target query $Y = q(\mathcal{M})$ by maximising the information gain from Eq. (3.5) given our hitherto collected data $\mathcal{D} := \boldsymbol{x}^{1:t-1}$. In Appx. C, we show that this is equivalent to maximising the following utility function:

$$U(a) = H(\boldsymbol{X}^t \mid \mathcal{D}) + \mathbb{E}_{\mathcal{M} \mid \mathcal{D}} \left[\mathbb{E}_{\boldsymbol{X}^t, Y \mid \mathcal{M}} \left[\log \mathbb{E}_{\mathcal{M}' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t, Y \mid \mathcal{M}') \right] \right] \right], \tag{4.11}$$

where

$$H(\boldsymbol{X}^t \mid \mathcal{D}) = \mathbb{E}_{\mathcal{M} \mid \mathcal{D}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log \mathbb{E}_{\mathcal{M}' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t \mid \mathcal{M}') \right] \right] \right]$$

denotes the differential entropy of the experiment outcome which depends on a and is distributed as in Eq. (3.6). This surrogate objective can be estimated using a nested Monte Carlo estimator as long as we can sample from and compute $p(Y \mid \mathcal{M})$, or alternatively, $p(Y \mid \boldsymbol{X}^t, G, \mathcal{D})$. Refer to Appx. C for further details. For example, for $q_{\mathrm{CR}}(\mathcal{M}) = p^{\mathrm{do}(X_i = \psi)}(X_j \mid \mathcal{M})$ with $\psi \sim p(\psi)$ a distribution over intervention values, we obtain:

$$\begin{aligned} U_{\mathrm{CR}}(a) = \mathbb{E}_{G \mid \mathcal{D}} \Big[& \mathbb{E}_{\boldsymbol{X}^t \mid G, \mathcal{D}} \Big[-\log \mathbb{E}_{G' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t \mid \mathcal{D}, G') \right] \\ & + \mathbb{E}_{\psi} \Big[\mathbb{E}^{\mathrm{do}(X_i = \psi)}_{X_j \mid \boldsymbol{X}^t, G, \mathcal{D}} \Big[\log \mathbb{E}_{G' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t \mid \mathcal{D}, G) \, p^{\mathrm{do}(X_i = \psi)}(X_j \mid \boldsymbol{X}^t, G, \mathcal{D}) \right] \Big] \Big] \Big] \Big]. \end{aligned} \tag{4.12}$$

Algorithm 1: GP-DiBS-ABCI for nonlinear additive Gaussian noise models

Input: # of experiments T, batch sizes $\{N_t\}_{t=1}^T$, # of latent particles M, # of MC samples K, particle resampling schedule $\{r_t\}_{t=1}^T$, hyperparameter update schedule $\{s_t\}_{t=1}^T$
Output: Posterior over target causal query $p(Y \mid \boldsymbol{x}^{1:T})$

$\boldsymbol{z}^0 \sim p(\boldsymbol{Z})$ ▷`sample initial particles;` Eq. (D.12)
for $t = 1$ **to** T **do**
  $a_t \leftarrow \arg\max_{a=(\mathcal{I},\boldsymbol{x}_\mathcal{I})} U(a, \boldsymbol{x}^{1:t-1})$ ▷`design experiment;` Eq. (4.11)
  $\boldsymbol{x}^t \leftarrow \{\boldsymbol{x}^{(t,n)} \sim p^{\mathrm{do}(a_t)}(\boldsymbol{X} \mid \mathcal{M}^\star)\}_{n=1}^{N_t}$ ▷`perform experiment;` Eq. (3.4)
  $\boldsymbol{z}^t \leftarrow \boldsymbol{z}^{t-1}$
  if r_t **then**
    $\boldsymbol{z}^t \leftarrow$ `resample_particles` $(\boldsymbol{z}^t)$ ▷`see Appx.E`
  end
  repeat
    $\boldsymbol{G} \leftarrow \{G^{(k,m)} \sim p(G \mid \boldsymbol{z}_m^t)\}_{k=1\ m=1}^{K\ \ M}$ ▷`sample graphs;` Eq. (D.11)
    $\boldsymbol{\kappa}, \boldsymbol{\sigma}^2 \leftarrow$ `estimate_hyperparameters`$(\boldsymbol{x}^{1:s_t}, \boldsymbol{G})$ ▷`see` Eq. (4.8)
    $\boldsymbol{z}^t \leftarrow$ `svgd_step`$(\boldsymbol{z}^t, \boldsymbol{x}^{1:t}, \boldsymbol{G}, \boldsymbol{\kappa}, \boldsymbol{\sigma}^2)$ ▷`update latent particles`
  until `svgd_convergence` ▷$\boldsymbol{z}^t$ `now approximate` $p(\boldsymbol{Z} \mid \boldsymbol{x}^{1:t})$
end

Importantly, for specific instances of the query function $q(\cdot)$ discussed in § 3, we can derive simpler utility functions than Eq. (4.11). For example, for $q_{\text{CD}}(\mathcal{M}) = G$ and $q_{\text{CML}}(\mathcal{M}) = \mathcal{M}$, we arrive at

$$U_{\text{CD}}(a) = \mathbb{E}_{G \mid \mathcal{D}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid G, \mathcal{D}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{D}, G) - \log \mathbb{E}_{G' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t \mid \mathcal{D}, G') \right] \right] \right], \tag{4.13}$$

$$U_{\text{CML}}(a) = \mathbb{E}_{\mathcal{M} \mid \mathcal{D}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}) - \log \mathbb{E}_{G' \mid \mathcal{D}} \left[p(\boldsymbol{X}^t \mid \mathcal{D}, G') \right] \right] \right], \tag{4.14}$$

where the entropy $\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} [\log p(\boldsymbol{X}^t \mid \mathcal{M})]$ can again be efficiently computed given our modelling choices. For brevity, we defer derivations and estimation details to Appxs. C and D.

Finding the optimal experiment $a_t^* = (\mathcal{I}^*, \boldsymbol{x}_\mathcal{I}^*)$ requires jointly optimising the utility function corresponding to our query with respect to (i) the set of intervention *targets* $\mathcal{I}$ and (ii) the corresponding intervention *values* $\boldsymbol{x}_\mathcal{I}$. This lends itself naturally to a nested, bi-level optimisation scheme [79]:

$$\mathcal{I}^* \in \arg\max_\mathcal{I} U(\mathcal{I}, \boldsymbol{x}_\mathcal{I}^*), \quad \text{where} \quad \forall \mathcal{I}: \quad \boldsymbol{x}_\mathcal{I}^* \in \arg\max_{\boldsymbol{x}_\mathcal{I}} U(\mathcal{I}, \boldsymbol{x}_\mathcal{I}), \tag{4.15}$$

In the above, we first estimate the optimal intervention values for all candidate intervention targets $\mathcal{I}$ and then select the intervention target that yields the highest utility. The intervention target $\mathcal{I}$ may contain multiple variables, which would yield a combinatorial problem. For simplicity, we consider only single-node interventions, $|\mathcal{I}| = 1$. To find $\boldsymbol{x}_\mathcal{I}^*$, we employ Bayesian optimisation [46, 47, 72] to efficiently estimate the most informative intervention value $\boldsymbol{x}_\mathcal{I}^*$, see Appx. D.

5 Experiments

Setup. We evaluate ABCI by inferring the query posterior on synthetic ground-truth SCMs using several different experiment selection strategies. Specifically, we design experiments w.r.t. U_{CD} (causal discovery), U_{CML} (causal model learning), and U_{CR} (causal reasoning); see § 4.2. We compare against baselines which (i) only sample from the observational distribution (OBS) or (ii) pick an intervention target j uniformly at random from $[d] \cup \{\varnothing\}$ and set $X_j = 0$ (RAND FIXED, a weak random baseline used in prior work) or draw $X_j \sim \mathcal{U}(-7, 7)$ (RAND) if $X_j \neq \varnothing$. All methods follow our Bayesian GP-DiBS-ABCI approach from § 4. We sample ground truth SCMs over random scale-free graphs [6] of size $d = 20$, with mechanisms and noise variances drawn from our model prior in Eq. (4.4). In Appx. G, we report additional results for both scale-free and Erdős Renyi random graphs over $d = 10$ resp. $d = 20$ variables. For specific prior choices and simulation details, see Appx. D.

Metrics. As ABCI infers a posterior over the target query Y, a natural evaluation metric is the Kullback-Leibler divergence (KLD) between the true query distribution and the inferred query posterior, $\mathrm{KL}(p(Y \mid \mathcal{M}^\star) \,||\, p(Y \mid \boldsymbol{x}^{1:t}))$. We report **Query KLD**, a KLD estimate for target interventional distributions (q_{CR}). As a proxy for the KLD of the SCM posterior (q_{CML}),[7] we report

[7] The SCM KLD is either zero, if the SCM posterior collapses onto the true SCM, or infinite, otherwise.

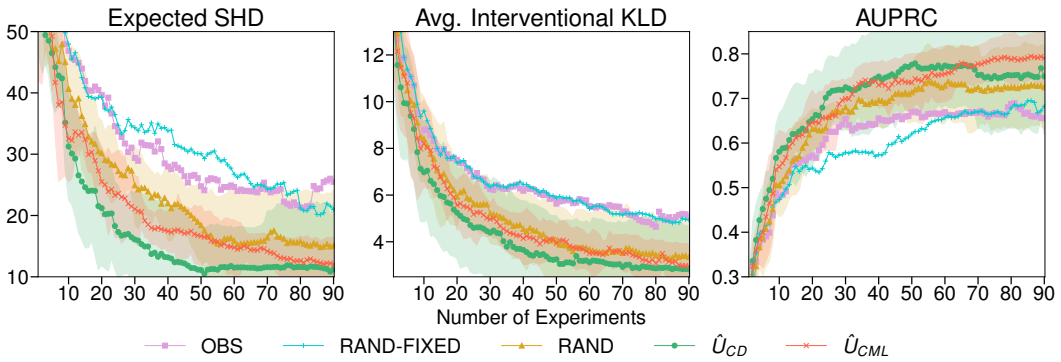

Figure 3: **Causal Discovery and SCM Learning.** Comparison of experimental design strategies for causal discovery (U_{CD}) and causal model learning (U_{CML}) with random and observational baselines on simulated ground truth models with 20 nodes. We initialise all methods with 50 observational samples, and then perform experiments with a batch size of $N_t = 5$. Lines and shaded areas show means and 95% confidence intervals (CIs) across 15 runs (5 randomly sampled ground-truth SCMs with 3 restarts per SCM). CIs for OBS and RAND FIXED baselines are not shown to aid readability; see Fig. 6 in Appx. G for the full figure. **(a) ESHD.** Both our objectives significantly outperform the observational and random baselines. **(b) Average I-KLD.** U_{CD} significantly outperforms the baselines, whereas U_{CML} performs only marginally better than RAND. **(c) AUPRC.** Both our strategies perform consistently better than the uninformed selection strategies.

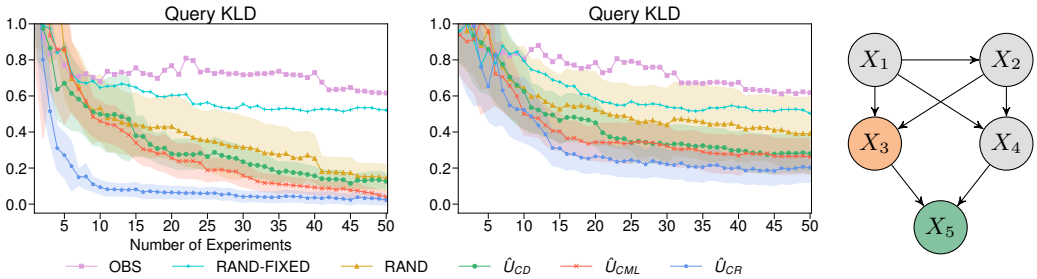

Figure 4: **Learning Interventional Distributions.** (left) Comparison of different methods w.r.t. learning a set of interventional distributions $p^{\text{do}(X_3=\psi)}(X_5 \mid \mathcal{M})$ with $\psi \sim \mathcal{U}[2, 5]$ on simulated ground truth models with fixed causal graph (right). We initialise all methods with 5 observational samples, and then perform experiments with a batch size of $N_t = 3$. Lines and shaded areas show means and 95% confidence intervals (CIs) across 30 runs (10 randomly sampled ground truth SCMs with 3 restarts each). CIs for OBS and RAND FIXED baselines are not shown to aid readability; see Figs. 9 and 10 in Appx. G for the full figure. **(a) All nodes actionable.** U_{CR} significantly outperforms all other methods as expected. U_{CML} performs second best which, in conjunction with the results from Fig. 3, suggests that U_{CML} yields a solid base model for performing downstream causal inference tasks. **(b)** X_3 **not actionable.** In this setting, where we cannot directly intervene on the treatment variable of interest, U_{CR} clearly outperforms all other methods for ≥ 10 experiments.

the average KLD across all single node interventional distributions $\{p^{\text{do}(X_i=\psi)}(\boldsymbol{X})\}_{i=1}^{d}$, with $\psi \sim \mathcal{U}(-7, 7)$ (**Average I-KLD**). We also report the expected structural Hamming distance [13], **ESHD** $= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}}[\text{SHD}(G, G^\star)]$, a commonly used causal discovery metric, and the *area under the precision recall curve* (**AUPRC**). See Appx. F for further details.

Causal Discovery and SCM Learning (Fig. 3). In our first experiment, we find that all ABCI-based methods are able to meaningfully learn from small amounts of data, which validates our Bayesian approach. Moreover, performing targeted interventions using experimental design indeed improves performance compared to uninformed experimentation (OBS, RAND FIXED, RAND). Notably, the stronger random baseline (RAND), which also randomises over intervention values, performs well in the considered setting. As expected by the theoretical grounding of the information gain utilities, U_{CD} identifies the true graph the fastest (as measured by ESHD), whereas U_{CML} exhibits good scores across all metrics. Further details are given in the caption of Fig. 3.

Learning Interventional Distributions (Fig. 4). In our second experiment, we investigate ABCI's causal reasoning capabilities by randomly sampling ground-truth SCMs as described above over the fixed graph shown in Fig. 4 (right), which is *not* known to the methods, and treat the (uncountable) set of interventional distributions $p^{\mathrm{do}(X_3=\psi)}(X_5 \mid \mathcal{M})$ with $\psi \sim \mathcal{U}[2, 5]$ as the target query. The results show that our informed experiment selection strategies significantly outperform the baselines at causal reasoning as measured by the Query KLD. In accordance with the results from Fig. 3 and considering that, once we know the true SCM, we can compute any causal quantity of interest, U_{CML} seems to provide a reasonable experimental strategy in case the causal query of interest is *not* known a priori. However, our results indicate that if we *do* know our query of interest, then U_{CR} provides a more efficient experiment design strategy for its estimation, even when the treatment variable of interest is not directly intervenable. In this case, the task is indeed more difficult, as highlighted by the larger Query KLD values across all considered methods.

6 Discussion

Assumptions, Limitations, and Extensions. In § 4, we have made several assumptions to facilitate tractable inference and showcase the ABCI framework in a relatively simple data-generating process. In particular, our assumptions exclude heteroscedastic noise, unobserved confounding, and cyclic relationships. On the experimental design side, we only considered hard interventions, but for some applications soft interventions [18] are more plausible. On the query side, we only considered interventional distributions. However, SCMs also naturally lend themselves to counterfactual reasoning, so one could also consider counterfactual queries such as the effect of the treatment on the treated [34, 71]. In principle, the ABCI framework as presented in § 3 extends directly to such generalisations. In practice, however, these can be non-trivial to implement, especially with regard to model parametrisation and tractable inference. Since actively performed interventions allow for causal learning even under causal sufficiency violations, we consider this a promising avenue for future work and believe the ABCI framework to be particularly well-suited for exploring it.

Reflections on the ABCI Framework. The main conceptual advantages of the ABCI framework are that it is *flexible* and *principled*. By considering general target causal queries, we can precisely specify what aspects of the causal model we are interested in. This conceptual framework offers a fresh perspective on the classical divide between causal discovery and reasoning: sometimes, the main objective may be to foster scientific understanding by uncovering the qualitative causal structure underlying real-world systems; other times, causal discovery may only be a means to an end to support causal reasoning. Of particular interest in the context of actively selecting interventions is the setting in which we cannot directly intervene on variables whose causal effect on others we are interested in (see Fig. 4), which connects to concepts such as transportability and external validity [7, 57]. ABCI is also flexible in that it easily allows for incorporating available domain knowledge: if we know some aspects of the model a priori (as assumed in conventional causal reasoning) [53] or have access to a large observational sample (from which we can infer the MEC of DAGs) [1], we can encode this in our prior and only optimise over a smaller model class. The principled Bayesian nature of ABCI comes at a significant computational cost: most integrals are intractable and approximating them with Monte-Carlo sampling is computationally expensive and can introduce bias when resources are limited, though cf. [84] for recent efforts to address such intractability. We discuss the computational complexity of our implementation in more detail in Appx. E.3. On the other hand, in many real-world applications, such as in the context of biological networks, active interventions are possible but only at a significant cost [11, 53]. In such cases in particular, a careful and computationally-heavy experimental design approach as presented in the present work is warranted and could be easily amortised.

Acknowledgments and Disclosure of Funding

We thank Paul K Rubenstein, Adrian Weller, and Bernhard Schölkopf for contributions to an early version of this work [79], and the anonymous reviewers for helpful feedback. This work was supported by: the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A, 01IS18039B; the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645; the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program grant agreement no. 815943; the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40 180545; and the Graz University of Technology LEAD project "Dependable Internet of Things in Adverse Environments".

References

[1] Agrawal, R., Squires, C., Yang, K., Shanmugam, K., and Uhler, C. (2019). ABCD-strategy: Budgeted experimental design for targeted causal structure discovery. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3400–3409. PMLR. 2, 3, 10, 18

[2] Agrawal, R., Uhler, C., and Broderick, T. (2018). Minimal I-MAP MCMC for scalable structure discovery in causal DAG models. In *International Conference on Machine Learning*, pages 89–98. PMLR. 2

[3] Angrist, J. D. and Pischke, J.-S. (2008). *Mostly Harmless Econometrics*. Princeton University Press. 1

[4] Annadani, Y., Rothfuss, J., Lacoste, A., Scherrer, N., Goyal, A., Bengio, Y., and Bauer, S. (2021). Variational causal networks: Approximate bayesian inference over causal structures. *arXiv preprint arXiv:2106.07635*. 2

[5] Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. (2020). BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*. 29

[6] Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512. 8, 24

[7] Bareinboim, E. and Pearl, J. (2016). Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences*, 113(27):7345–7352. 10

[8] Beck, J., Dia, B. M., Espath, L. F., Long, Q., and Tempone, R. (2018). Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain. *Computer Methods in Applied Mechanics and Engineering*, 334. 28

[9] Brouillard, P., Lachapelle, S., Lacoste, A., Lacoste-Julien, S., and Drouin, A. (2020). Differentiable Causal Discovery from Interventional Data. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21865–21877. Curran Associates, Inc. 2

[10] Chaloner, K. and Verdinelli, I. (1995). Bayesian experimental design: A review. *Statistical Science*, pages 273–304. 2, 5

[11] Cho, H., Berger, B., and Peng, J. (2016). Reconstructing causal biological networks through active learning. *PloS one*, 11(3):e0150611. 2, 10, 18

[12] Cundy, C., Grover, A., and Ermon, S. (2021). BCD nets: Scalable variational approaches for Bayesian causal discovery. *Advances in Neural Information Processing Systems*, 34. 2

[13] de Jongh, M. and Druzdzel, M. J. (2009). A comparison of structural distance measures for causal bayesian network models. *Recent Advances in Intelligent Information Systems, Challenging Problems of Science, Computer Science series*, pages 443–456. 9

[14] Deleu, T., Góis, A., Emezue, C. C., Rankawat, M., Lacoste-Julien, S., Bauer, S., and Bengio, Y. (2022). Bayesian structure learning with generative flow networks. In *The 38th Conference on Uncertainty in Artificial Intelligence*. 2

[15] Eberhardt, F. (2008). Almost optimal intervention sets for causal discovery. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, pages 161–168. AUAI Press. 2

[16] Eberhardt, F., Glymour, C., and Scheines, R. (2005). On the number of experiments sufficient and in the worst case necessary to identify all causal relations among n variables. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*. 2

[17] Eberhardt, F., Glymour, C., and Scheines, R. (2006). N-1 experiments suffice to determine the causal relations among n variables. In *Innovations in machine learning*, pages 97–112. Springer. 2

[18] Eberhardt, F. and Scheines, R. (2007). Interventions and causal inference. *Philosophy of Science*, 74(5):981–995. 10

[19] Ellis, B. and Wong, W. H. (2008). Learning causal bayesian network structures from experimental data. *Journal of the American Statistical Association*, 103(482):778–789. 31

[20] Erdös, P. and Rényi, A. (1959). On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290. 25

[21] Friedman, N. and Koller, D. (2003). Being Bayesian about network structure. a Bayesian approach to structure discovery in Bayesian networks. *Machine learning*, 50(1):95–125. 2, 31

[22] Friedman, N. and Nachman, I. (2000). Gaussian process networks. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 211–219. Morgan Kaufmann Publishers Inc. 2

[23] Gamella, J. L. and Heinze-Deml, C. (2020). Active invariant causal prediction: Experiment selection through stability. *Advances in Neural Information Processing Systems*, 33:15464–15475. 2

[24] Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., and Wilson, A. G. (2018). Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*. 29

[25] George Casella, R. L. B. (2002). *Statistical Inference*, volume 2. Duxbury. 4

[26] Ghassami, A., Salehkaleybar, S., Kiyavash, N., and Bareinboim, E. (2018). Budgeted Experiment Design for Causal Structure Learning. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1724–1733. PMLR. 2

[27] Glymour, C., Zhang, K., and Spirtes, P. (2019). Review of Causal Discovery Methods Based on Graphical Models. *Frontiers in Genetics*, 10. 2

[28] Goda, T., Hironaka, T., and Iwamoto, T. (2020). Multilevel Monte Carlo estimation of expected information gains. *Stochastic Analysis and Applications*, 38. 28

[29] Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring Network Structure, Dynamics, and Function using NetworkX. *Proceedings of the 7th Python in Science Conference*, pages 11–15. 24, 29

[30] Hauser, A. and Bühlmann, P. (2012). Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1):2409–2464. 2

[31] Hauser, A. and Bühlmann, P. (2014). Two optimal strategies for active learning of causal models from interventional data. *International Journal of Approximate Reasoning*, 55(4):926–939. 2

[32] He, Y.-B. and Geng, Z. (2008). Active learning of causal networks with intervention experiments and optimal designs. *Journal of Machine Learning Research*, 9(Nov):2523–2547. 2

[33] Heckerman, D. (1995). A Bayesian approach to learning causal networks. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 285–295. Morgan Kaufmann Publishers Inc. 2

[34] Heckman, J. J. (1992). Policy evaluation. *Evaluating welfare and training programs*, page 201. 10

[35] Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. (2018). Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391. 2

[36] Hernán, M. A. and Robins, J. M. (2020). *Causal Inference: What If*. Boca Raton: Chapman & Hall/CRC. 1

[37] Hoyer, P. O., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. (2009). Nonlinear causal discovery with additive noise models. In *Advances in neural information processing systems*, pages 689–696. 1, 2, 5

[38] Hyttinen, A., Eberhardt, F., and Hoyer, P. O. (2013). Experiment selection for causal discovery. *The Journal of Machine Learning Research*, 14(1):3041–3071. 2

[39] Imbens, G. W. and Rubin, D. B. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press. 1

[40] Jaber, A., Kocaoglu, M., Shanmugam, K., and Bareinboim, E. (2020). Causal discovery from soft interventions with unknown targets: Characterization and learning. *Advances in neural information processing systems*, 33:9551–9561. 2

[41] Kalainathan, D., Goudet, O., and Dutta, R. (2020). Causal discovery toolbox: Uncovering causal relationships in python. *Journal of Machine Learning Research*, 21(37). 29

[42] Lindley, D. V. et al. (1956). On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005. 2, 5

[43] Lippe, P., Cohen, T., and Gavves, E. (2021). Efficient neural causal discovery without acyclicity constraints. In *International Conference on Learning Representations*. 2

[44] Liu, Q. and Wang, D. (2016). Stein variational gradient descent: A general purpose Bayesian inference algorithm. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc. 7, 25

[45] Lorch, L., Rothfuss, J., Schölkopf, B., and Krause, A. (2021). DiBS: Differentiable Bayesian Structure Learning. *Advances in Neural Information Processing Systems*, 34. 2, 7, 25, 31

[46] Mockus, J. (1975). On Bayesian methods for seeking the extremum. In *Optimization Techniques IFIP Technical Conference*, pages 400–404. Springer. 2, 8, 26

[47] Mockus, J. (2012). *Bayesian Approach to Global Optimization: Theory and Applications*, volume 37. Springer Science & Business Media. 2, 8, 26

[48] Mooij, J. M., Peters, J., Janzing, D., Zscheischler, J., and Schölkopf, B. (2016). Distinguishing cause from effect using observational data: methods and benchmarks. *The Journal of Machine Learning Research*, 17(1):1103–1204. 1

[49] Morgan, S. L. and Winship, C. (2014). *Counterfactuals and Causal Inference: Methods and Principles for Social Research*. Cambridge University Press. 1

[50] Murphy, K. P. (2001). Active learning of causal Bayes net structure. Technical report, Department of Computer Science, U.C. Berkeley. 2, 18

[51] Murphy, K. P. (2007). Conjugate Bayesian analysis of the gaussian distribution. Technical report, University of British Columbia. 6, 25

[52] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press. 31

[53] Ness, R. O., Sachs, K., Mallick, P., and Vitek, O. (2017). A Bayesian active learning experimental design for inferring signaling networks. In *International Conference on Research in Computational Molecular Biology*, pages 134–156. Springer. 2, 10

[54] 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. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc. 29

[55] Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, 82(4):669–688. 1

[56] Pearl, J. (2009). *Causality*. Cambridge University Press, 2nd edition. 1, 2, 3, 4

[57] Pearl, J. and Bareinboim, E. (2014). External validity: From do-calculus to transportability across populations. *Statistical Science*, 29(4):579–595. 10

[58] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830. 29

[59] Peters, J., Bühlmann, P., and Meinshausen, N. (2016). Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012. 2

[60] Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of Causal Inference - Foundations and Learning Algorithms*. Adaptive Computation and Machine Learning Series. The MIT Press, Cambridge, MA, USA. 1, 2, 6

[61] Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15(1):2009–2053. 5

[62] Rainforth, T., Cornish, R., Yang, H., Warrington, A., and Wood, F. (2018). On nesting Monte Carlo estimators. *35th International Conference on Machine Learning, ICML 2018*, 10. 28

[63] Robins, J. (1986). A new approach to causal inference in mortality studies with a sustained exposure period—application to control of the healthy worker survivor effect. *Mathematical modelling*, 7(9-12):1393–1512. 5

[64] Robinson, R. W. (1973). Counting labeled acyclic digraphs. *New Directions in the Theory of Graphs*, pages 239–273. 7

[65] Rubenstein, P. K., Tolstikhin, I., Hennig, P., and Schölkopf, B. (2017). Probabilistic active learning of functions in structural causal models. *arXiv preprint arXiv:1706.10234*. 3

[66] Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331. 2

[67] Scherrer, N., Bilaniuk, O., Annadani, Y., Goyal, A., Schwab, P., Schölkopf, B., Mozer, M. C., Bengio, Y., Bauer, S., and Ke, N. R. (2021). Learning neural causal models with active interventions. *arXiv preprint arXiv:2109.02429*. 2

[68] Shalit, U., Johansson, F. D., and Sontag, D. (2017). Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*, pages 3076–3085. PMLR. 2

[69] Shanmugam, K., Kocaoglu, M., Dimakis, A. G., and Vishwanath, S. (2015). Learning Causal Graphs with Small Interventions. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc. 2, 18

[70] Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M. (2006). A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10). 1, 2

[71] Shpitser, I. and Pearl, J. (2009). Effects of treatment on the treated: Identification and generalization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2009*, pages 514–521. AUAI Press. 10

[72] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959. 2, 8

[73] Soch, J., Faulkenberry, Thomas Petrykowski, J., Allefeld, C., and McInerney, C. D. (2022). *The Book of Statistical Proofs*. Zenodo. 22

[74] Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, prediction, and search*. MIT press, 2nd edition. 1, 2, 5

[75] Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. In *International Conference on Machine Learning*. 26

[76] Sussex, S., Uhler, C., and Krause, A. (2021). Near-optimal multi-perturbation experimental design for causal structure learning. *Advances in Neural Information Processing Systems*, 34. 2

[77] Tigas, P., Annadani, Y., Jesson, A., Schölkopf, B., Gal, Y., and Bauer, S. (2022). Interventions, Where and How? Experimental Design for Causal Models at Scale. *arXiv preprint arXiv:2203.02016*. 2, 3, 18, 31

[78] Tong, S. and Koller, D. (2001). Active learning for structure in Bayesian networks. In *International Joint Conference on Artificial Intelligence*, volume 17, pages 863–869. 2, 18

[79] von Kügelgen, J., Rubenstein, P. K., Schölkopf, B., and Weller, A. (2019). Optimal experimental design via Bayesian optimization: active causal structure learning for Gaussian process networks. In *NeurIPS 2019 Workshop "Do the right thing": machine learning and causal inference for improved decision making*. *arXiv:1910.03962*. 2, 8, 10, 26

[80] Vowels, M. J., Camgoz, N. C., and Bowden, R. (2022). D'ya Like DAGs? A Survey on Structure Learning and Causal Discovery. *ACM Computing Surveys*. 2

[81] Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian Processes for Machine Learning*, volume 2. MIT Press Cambridge, MA. 2, 6, 7, 19

[82] Wright, S. (1934). The method of path coefficients. *The Annals of Mathematical Statistics*, 5(3):161–215. 1

[83] Yang, K., Katcoff, A., and Uhler, C. (2018). Characterizing and learning equivalence classes of causal dags under interventions. In *International Conference on Machine Learning*, pages 5541–5550. PMLR. 2

[84] Zemplenyi, M. and Miller, J. W. (2022). Bayesian optimal experimental design for inferring causal structure. *Bayesian Analysis*, 1(1):1–28. 10

[85] Zhang, K. and Hyvärinen, A. (2009). On the identifiability of the post-nonlinear causal model. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 647–655. AUAI Press. 1

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Summarising the abstract/introduction we claim (i) to introduce *a principled fully-Bayesian active learning framework for integrated causal discovery and reasoning* and to (ii) show the practicality of our approach through simulations. We lay out the former concisely in § 3 and § 4. We report the empirical evaluation in § 5.
   (b) Did you describe the limitations of your work? [Yes] See discussion in § 6.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] We give a concise and rigorous treatment when formulating the general framework in § 3, as well as our approach and model specifics in § 4.
   (b) Did you include complete proofs of all theoretical results? [Yes] We provide the derivation of our utility functions in Appx. C.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Python code and instructions are provided in the supplement as source_code.zip
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] We give a minimal set of details in § 5 and provide full information about our experiments in Appxs. D to F.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Figs. 3 and 4
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We give a brief summary in Appendix E.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We do not use external models or data. We use a set of Python packages that we list in Appendix E.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Our code base is available on Github at https://www.github.com/chritoth/active-bayesian-causal-inference. We provide this link on the first page of the main paper and in Appx. E.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendices

Table of Contents

A Further Discussion of Related Work **18**

B Background on Gaussian processes **19**

C Derivation of the Information Gain Utility Functions **20**
- C.1 Information Gain for General Queries 20
- C.2 Derivation of the Causal Discovery Utility Function 20
- C.3 Derivation of the Causal Model Learning Utility Function 21
- C.4 Derivation of the Causal Reasoning Utility Function 23

D Approximate Inference and Experimental Details **24**
- D.1 Estimating Posterior Marginal Interventional Likelihoods 24
- D.2 Sampling Ground Truth Graphs 24
- D.3 Normal-Inverse-Gamma Prior for Root Nodes 25
- D.4 Gamma Priors for GP Hyperparameters of Non-Root Nodes 25
- D.5 DiBS for Approximate Posterior Graph Inference 25
- D.6 Estimation of the Information Gain Utility Functions 25
- D.7 Bayesian Optimisation for Experimental Design 26

E Implementation Details **27**
- E.1 Particle Resampling 27
- E.2 Shared Priors and Caching of Marginal Likelihoods 27
- E.3 Computational Complexity 28
- E.4 Implementation and Computing Resources 29

F Evaluation Metrics **30**

G Extended Experimental Results **32**

A Further Discussion of Related Work

In this section, we further discuss the most closely related prior works, which also consider a Bayesian active learning approach for causal discovery. These methods are summarised and contrasted with ABCI in Tab. 1. Similar to our approach, they also all assume acyclicity and causal sufficiency.

Table 1: Comparison of ABCI with closely related active Bayesian causal discovery methods in terms of the learning objective, that is, the causal target query, and the considered model class.

Work	**Target Query**	**Model Class**
Tong and Koller [78], Murphy [50]	causal graph G	Conjugate Dirichlet-Multinomial
Cho et al. [11]	causal graph G	Conjugate linear Gaussian-inverse-Gamma
Agrawal et al. [1]	some function $\phi(G)$ of the causal graph G	Linear Gaussian
Tigas et al. [77]	causal graph G and parameters of f_i	Additive Gaussian noise with parametric neural network functions f_i
GP-DiBS-ABCI (ours)	some function $q(\mathcal{M})$ of the full SCM $\mathcal{M}$	Additive Gaussian noise with nonparametric functions f_i modeled by GPs

The early experimental design work by Tong and Koller [78] and Murphy [50] already investigated active causal discovery from a Bayesian perspective. They focused on the case in which all variables are multinomial to allow for tractable, closed-form posterior inference with a conjugate Dirichlet prior.

The setting with continuous variables was not explored from an active Bayesian causal discovery perspective until the work of Cho et al. [11], who consider the linear Gaussian case in the context of biological networks. Cho et al. [11] similarly use an inverse-Gamma prior to enable closed-form posterior inference. In these approaches, experiment selection targets the full causal graph. Agrawal et al. [1] extend the work of Cho et al. [11] by enabling the active learning of some function of the causal graph and handling interventional budget constraints.

Similarly to our approach, the concurrent work by Tigas et al. [77] models nonlinear causal relationships with additive Gaussian noise in the active learning setting. However, they are limited to targeting the full SCM for experiment design, which corresponds to our q_{CML} objective. In addition, their approach does not quantify the uncertainty in the functions conditional on a causal graph sampled from the graph posterior. In contrast, our nonparametric approach both directly models the epistemic uncertainty in the functions and mitigates the risk of model misspecification by jointly learning the kernel hyperparameters. Moreover, our method is Bayesian over the unknown noise variances, which are usually unknown in practice. It is unclear whether Tigas et al. [77] hand-specify a constant noise variance a priori, or whether they infer it jointly with the function parameters [77, cf. § 5.4.1].

Other related work by Shanmugam et al. [69] considers the problem of finding the minimal number of perfectly informative (w.r.t. conditional independences induced by the true underlying graph) multi-target interventions to fully identify the true causal graph. In contrast, we assume that only finitely many data points are available per experiment/intervention. Thus, we try to perform at each time step (possibly repeating) interventions to maximally reduce our uncertainty in the target causal query (which may be the causal graph). As another point of difference, we also optimise for the actual intervention value, whereas Shanmugam et al. [69] only optimise for the intervention targets.

B Background on Gaussian processes

We use Gaussian Processes (GPs) to model mechanisms of non-root nodes X_i, i.e., we place a GP prior on $p(f_i \mid G)$. In the following, we give some background on GPs and how to compute probabilistic quantities thereof relevant to this work. For further information on GPs we refer the reader to Williams and Rasmussen [81].

A $\mathcal{GP}(m_i(\cdot), k_i^G(\cdot,\cdot))$ is a collection of random variables, any finite number of which have a joint Gaussian distribution, and is fully determined by its mean function $m_i(\cdot)$ and covariance function (or kernel) $k_i^G(\cdot,\cdot)$, where

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})], \quad \text{and} \quad k(\mathbf{x},\mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]. \tag{B.1}$$

In our experiments, we choose the mean function $m_i(x) \equiv 0$ to be zero and a rational quadratic kernel

$$k_{RQ}(\mathbf{x},\mathbf{x}') = \kappa_i^o \cdot \left(1 + \frac{1}{2\alpha}(\mathbf{x}-\mathbf{x}')^\top \kappa_i^l (\mathbf{x}-\mathbf{x}')\right)^{-\alpha} \tag{B.2}$$

as our covariance function. Here, α denotes a weighting parameter, κ_i^o denotes an output scale parameter and κ_i^l denotes a length scale parameter. For the weighting parameter, we use a default value of $\alpha = \log 2 \approx 0.693$. For κ_i^l and κ_i^o we choose priors according to Appx. D.4. In Section 4.1 we summarise both parameters as $\boldsymbol{\kappa}_i = (\kappa_i^o, \kappa_i^l)$.

In this work, we consider Gaussian additive noise models (see Eq. (4.1)). Hence, for a given non-root node X_i in some graph G, we have

$$p(X_i \mid \mathbf{pa}_i^G, f_i, \sigma_i^2, G) = \mathcal{N}(X_i \mid f_i(\mathbf{pa}_i^G), \sigma_i^2) \tag{B.3}$$

where $\mathbf{pa}_i^G$ denotes the parents of X_i in G. For some batch of collected data $\boldsymbol{x} = \{\boldsymbol{x}^n\}_{n=1}^N$, let $\boldsymbol{x}_i = (x_i^1, \dots x_i^N)^T$, $\mathbf{pa}_i^G = (\mathbf{pa}_i^{G,1}, \dots, \mathbf{pa}_i^{G,N})$, and $\boldsymbol{K}$ the Gram matrix with entries $K_{m,n} = k_{RQ}(\mathbf{pa}_i^{G,m}, \mathbf{pa}_i^{G,n})$. Then, we can compute the prior marginal log-likelihood, which is needed to compute $p(x^{1:t} \mid G)$, in closed form as

$$\log p(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, \sigma_i^2, G) = \log \mathbb{E}_{f_i \mid G}\left[p(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, f_i, \sigma_i^2, G)\right] \tag{B.4}$$

$$= -\frac{1}{2}\boldsymbol{x}_i^T (K + \sigma^2 I)^{-1} \boldsymbol{x}_i - \frac{1}{2}\log|K + \sigma^2 I| - \frac{N}{2}\log 2\pi. \tag{B.5}$$

To predict the function values $f_i(\widetilde{\mathbf{pa}}_i^G)$ at unseen test locations $\widetilde{\mathbf{pa}}_i^G = (\widetilde{\mathbf{pa}}_i^{G,1}, \dots, \widetilde{\mathbf{pa}}_i^{G,\tilde{N}})$ given previously observed data $\boldsymbol{x}$, let $\boldsymbol{K}^\dagger$ be the $(\tilde{N} \times N)$ covariance matrix with entries $K^\dagger_{m,n} = k_{RQ}(\widetilde{\mathbf{pa}}_i^{G,m}, \mathbf{pa}_i^{G,n})$ and $\tilde{\boldsymbol{K}}$ be the $(\tilde{N} \times \tilde{N})$ covariance matrix with entries $\tilde{K}_{m,n} = k_{RQ}(\widetilde{\mathbf{pa}}_i^{G,m}, \widetilde{\mathbf{pa}}_i^{G,n})$. Then, the predictive posterior is multivariate Gaussian

$$p(f_i(\widetilde{\mathbf{pa}}_i^G) \mid \widetilde{\mathbf{pa}}_i^G, \boldsymbol{x}, \sigma_i^2, G) = \mathcal{N}(\boldsymbol{\mu}_f, \boldsymbol{\Sigma}_f) \tag{B.6}$$

with mean

$$\boldsymbol{\mu}_f = \boldsymbol{K}^\dagger \left[\boldsymbol{K} + \sigma_i^2 I\right]^{-1} \boldsymbol{x}_i \tag{B.7}$$

and covariance

$$\boldsymbol{\Sigma}_f = \tilde{\boldsymbol{K}} - \boldsymbol{K}^\dagger \left[\boldsymbol{K} + \sigma_i^2 I\right]^{-1} \boldsymbol{K}^\dagger. \tag{B.8}$$

Finally, the marginal posterior over observations $\tilde{\boldsymbol{X}}_i$, which is needed to sample and evaluate candidate experiments in the experimental design process, is given by

$$p(\tilde{\boldsymbol{X}}_i \mid \widetilde{\mathbf{pa}}_i^G, \boldsymbol{x}, \sigma_i^2, G) = \mathcal{N}(\boldsymbol{\mu}_{X_i}, \boldsymbol{\Sigma}_{X_i}) \tag{B.9}$$

with mean

$$\boldsymbol{\mu}_{X_i} = \boldsymbol{\mu}_f \tag{B.10}$$

and covariance

$$\boldsymbol{\Sigma}_{X_i} = \boldsymbol{\Sigma}_f + \sigma_i^2 I. \tag{B.11}$$

C Derivation of the Information Gain Utility Functions

In the following, we provide the derivations for the expressions presented in Section 4.2.

C.1 Information Gain for General Queries

We show that

$$\arg\max_{a_t} \ \mathrm{I}(Y; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) \ = \ \arg\max_{a_t} \ U(a_t) \tag{C.1}$$

for $U(a_t)$ given in Eq. (4.11).

Proof. We write the mutual information in the following form

$$\mathrm{I}(Y; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) = H(Y \mid \boldsymbol{x}^{1:t-1}) + H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) - H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}). \tag{C.2}$$

In the above, we expand the joint entropy of experiment outcome and query as

$$\begin{aligned}
H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) &= -\mathbb{E}_{Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right] && \text{(C.3)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{Y, \boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right]\right] && \text{(C.4)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{Y, \boldsymbol{X}^t \mid \mathcal{M}} \left[\log \mathbb{E}_{\mathcal{M}' \mid \boldsymbol{x}^{1:t-1}} \left[p(Y \mid \mathcal{M}') \cdot p(\boldsymbol{X}^t \mid \mathcal{M}')\right]\right]\right] && \text{(C.5)}
\end{aligned}$$

for any query such that query and experiment outcome are conditionally independent given an SCM. This holds true, e.g., whenever Y is a deterministic function of $\mathcal{M}$ such as $Y = q_{\text{CD}}(\mathcal{M}) = G$.

The marginal entropy of the experiment outcome given previously observed data is

$$\begin{aligned}
H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) &= -\mathbb{E}_{\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right] && \text{(C.6)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right]\right] && \text{(C.7)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log \mathbb{E}_{\mathcal{M}' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid \mathcal{M}')\right]\right]\right] && \text{(C.8)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log \mathbb{E}_{\boldsymbol{f}', \boldsymbol{\sigma}^{2'}, G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid \boldsymbol{f}', \boldsymbol{\sigma}^{2'}, G')\right]\right]\right] && \text{(C.9)}\\
&= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\right]\right] && \text{(C.10)}\\
&= -\mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2, G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \boldsymbol{f}, \boldsymbol{\sigma}^2, G} \left[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\right]\right] && \text{(C.11)}\\
&= -\mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}} \left[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\right]\right] && \text{(C.12)}
\end{aligned}$$

Finally, since the query posterior entropy $H(Y \mid \boldsymbol{x}^{1:t-1})$ does not depend on the candidate experiment a_t, we obtain

$$\begin{aligned}
&\arg\max_{a_t} \quad \mathrm{I}(Y; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\\
=&\arg\max_{a_t} \quad H(Y \mid \boldsymbol{x}^{1:t-1}) + H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) - H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\\
=&\arg\max_{a_t} \quad H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) - H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) && \text{(C.13)}
\end{aligned}$$

which, together with Eqs. (C.5) and (C.8), completes the proof. □

C.2 Derivation of the Causal Discovery Utility Function

To derive $U_{\text{CD}}(a)$, we note that $Y = q_{\text{CD}}(\mathcal{M}) = G$, and hence the joint entropy of experiment outcome and query in Eq. (C.3) becomes

$$\begin{aligned}
H(G, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) &= -\mathbb{E}_{G, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(G, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right] && \text{(C.14)}\\
&= -\mathbb{E}_{G, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}) + \log p(G \mid \boldsymbol{x}^{1:t-1})\right] && \text{(C.15)}\\
&= -\mathbb{E}_{G, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1})\right] + H(G \mid \boldsymbol{x}^{1:t-1}) && \text{(C.16)}\\
&= -\mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1})\right]\right] + H(G \mid \boldsymbol{x}^{1:t-1}). && \text{(C.17)}
\end{aligned}$$

Substituting this into Eq. (C.2) yields

$$\mathrm{I}(G; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) \tag{C.18}$$
$$= H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) + \mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid G,\, \boldsymbol{x}^{1:t-1}) \right] \right]. \tag{C.19}$$

By Eq. (C.12), we have

$$= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid G,\, \boldsymbol{x}^{1:t-1}) - \log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1}) \right] \right] \right] \tag{C.20}$$

which recovers the utility function $U_{\text{CD}}(a)$ from Eq. (4.13).

C.3 Derivation of the Causal Model Learning Utility Function

To derive $U_{\text{CML}}(a)$ given $Y = q_{\text{CML}}(\mathcal{M}) = \mathcal{M}$, the joint entropy of experiment outcome and query in Eq. (C.3) are given by

$$H(\mathcal{M},\, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) = -\mathbb{E}_{\mathcal{M},\, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\mathcal{M},\, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) \right] \tag{C.21}$$
$$= -\mathbb{E}_{\mathcal{M},\, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}, \boldsymbol{x}^{1:t-1}) + \log p(\mathcal{M} \mid \boldsymbol{x}^{1:t-1}) \right] \tag{C.22}$$
$$= -\mathbb{E}_{\mathcal{M},\, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}) \right] + H(\mathcal{M} \mid \boldsymbol{x}^{1:t-1}) \tag{C.23}$$
$$= -\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}) \right] \right] + H(\mathcal{M} \mid \boldsymbol{x}^{1:t-1}). \tag{C.24}$$

As previously, substituting this into Eq. (C.2) yields

$$\mathrm{I}(G; \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) = H(\boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) + \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M},) \right] \right] \tag{C.25}$$

and by Eq. (C.10), we have

$$= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}) - \log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}} \left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1}) \right] \right] \right] \tag{C.26}$$

which recovers the utility $U_{\text{CML}}(a)$ from Eq. (4.14).

For our concrete modeling choices we can further simplify this utility. Let $\mathbf{Anc}_i^{\mathcal{M}}$ and $\mathbf{Pa}_i^{\mathcal{M}}$ denote the ancestor and parent sets of node X_i in $\mathcal{M}$. Then,

$$\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p(\boldsymbol{X}^t \mid \mathcal{M}) \right] \right] \tag{C.27}$$
$$= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log \prod_{i \notin \mathcal{I}^t} p^{\text{do}(a_t)}(\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \mathcal{M}) \right] \right] \tag{C.28}$$
$$= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\sum_{i \notin \mathcal{I}^t} \log p^{\text{do}(a_t)}(\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \mathcal{M}) \right] \right] \tag{C.29}$$
$$= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} \mathbb{E}_{\boldsymbol{X}^t \mid \mathcal{M}} \left[\log p^{\text{do}(a_t)}(\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \mathcal{M}) \right] \right] \tag{C.30}$$
$$= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} \mathbb{E}_{\mathbf{Anc}_i^{\mathcal{M}} \mid \text{do}(a_t), \mathcal{M}} \left[\mathbb{E}_{\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \text{do}(a_t), \mathcal{M}} \left[\log p^{\text{do}(a_t)}(\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \mathcal{M}) \right] \right] \right]. \tag{C.31}$$

Since our root nodes and GPs assume an additive Gaussian noise model, the innermost expectation amounts to the negative entropy the Gaussian noise variable, i.e.,

$$\mathbb{E}_{\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \text{do}(a_t), \mathcal{M}} \left[\log p^{\text{do}(a_t)}(\boldsymbol{X}_i^t \mid \mathbf{pa}_i^{\mathcal{M}}, \mathcal{M}) \right] = -\frac{N_t}{2} \log(2\pi\sigma_i^2 e). \tag{C.32}$$

As we further assume a homoscedastic noise model for our GPs, Eq. (C.31) reduces to

$$\mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} -\frac{N_t}{2} \log(2\pi\sigma_i^2 e) \right] \tag{C.33}$$

$$= -\mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2, G \mid \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} \frac{N_t}{2} \log(2\pi\sigma_i^2 e) \right] \tag{C.34}$$

$$= -\mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\mathbb{E}_{\boldsymbol{\sigma}^2 \mid G, \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} \frac{N_t}{2} \log(2\pi\sigma_i^2 e) \right] \right] \tag{C.35}$$

$$= -\mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}} \left[\sum_{i \notin \mathcal{I}^t} \mathbb{E}_{\sigma_i^2 \mid G, \boldsymbol{x}^{1:t-1}} \left[\frac{N_t}{2} \log(2\pi\sigma_i^2 e) \right] \right], \tag{C.36}$$

which can be approximated by nested Monte Carlo estimation. For non-root nodes we approximate the inner expectation with a single point estimate (cf. Section 4.1). For root nodes we can compute the inner expectation in closed form as

$$\mathbb{E}_{\sigma_i^2 \mid G, \boldsymbol{x}^{1:t-1}} \left[\frac{N_t}{2} \log(2\pi\sigma_i^2 e) \right] = \frac{N_t}{2} \left(\log(2\pi e) - \psi(\alpha_i^t) + \log \beta_i^t \right) \tag{C.37}$$

where α_i^t, β_i^t are the parameters of the inverse-gamma noise posterior $\sigma_i^2 \sim \Gamma^{-1}(\sigma_i^2 \mid \alpha_i^t, \beta_i^t)$ (see Appx. D.3) and $\psi(\cdot)$ is the digamma function.

Proof (adapted from [73]). We need to show that

$$\mathbb{E}_{\sigma^2} \left[\log(\sigma^2) \right] = -\psi(\alpha) + \log \beta \tag{C.38}$$

where the noise variance σ^2 follows an inverse-gamma density

$$\sigma^2 \sim \Gamma^{-1}(\sigma^2 \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \cdot (\sigma^2)^{-\alpha-1} \cdot e^{-\frac{\beta}{\sigma^2}}. \tag{C.39}$$

By substituting $y = \log \sigma^2$ we get

$$y \sim p(y \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \cdot e^{-\alpha y} \cdot e^{-\beta e^{-y}}. \tag{C.40}$$

Now note that

$$\int_{-\infty}^{\infty} p(y \mid \alpha, \beta) dy = 1 \tag{C.41}$$

and hence

$$\frac{\Gamma(\alpha)}{\beta^\alpha} = \int_{-\infty}^{\infty} e^{-\alpha y} \cdot e^{-\beta e^{-y}} dy. \tag{C.42}$$

By differentiating the latter integrand w.r.t. α we get

$$\frac{d}{d\alpha} \left(e^{-\alpha y} \cdot e^{-\beta e^{-y}} \right) = (-y) e^{-\alpha y} \cdot e^{-\beta e^{-y}} = (-y) \cdot p(y \mid \alpha, \beta) \cdot \frac{\Gamma(\alpha)}{\beta^\alpha}. \tag{C.43}$$

Bringing the parts together we obtain

$$\mathbb{E}_{\sigma^2}\left[\log(\sigma^2)\right] = \mathbb{E}_y\left[y\right] \tag{C.44}$$

$$= \int_{-\infty}^{\infty} y \cdot p(y \mid \alpha, \beta) dy \tag{C.45}$$

$$\overset{Eq.\ (C.43)}{=} -\frac{\beta^\alpha}{\Gamma(\alpha)} \int_{-\infty}^{\infty} \frac{d}{d\alpha}\left(e^{-\alpha y} \cdot e^{-\beta e^{-y}}\right) dy \tag{C.46}$$

$$= -\frac{\beta^\alpha}{\Gamma(\alpha)} \frac{d}{d\alpha}\left(\int_{-\infty}^{\infty} e^{-\alpha y} \cdot e^{-\beta e^{-y}} dy\right) \tag{C.47}$$

$$\overset{Eq.\ (C.42)}{=} -\frac{\beta^\alpha}{\Gamma(\alpha)} \frac{d}{d\alpha}\left(\frac{\Gamma(\alpha)}{\beta^\alpha}\right) \tag{C.48}$$

$$= -\frac{\beta^\alpha}{\Gamma(\alpha)}\left(\beta^{-\alpha} \cdot \frac{d}{d\alpha}\Gamma(\alpha) - \Gamma(\alpha) \cdot \beta^{-\alpha} \cdot \log\beta\right) \tag{C.49}$$

$$= -\psi(\alpha) + \log\beta, \tag{C.50}$$

which completes the proof. □

In summary, in our instance of GP-DIBS-ABCI we estimate the causal model learning utility as

$$U_{\text{CML}}(a_t) = -\mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}}\Bigg[\sum_{i \in \mathbf{R}(G) \setminus \mathcal{I}^t} \frac{N_t}{2}\left(\log(2\pi e) - \psi(\alpha_i^t) + \log\beta_i^t\right) + \sum_{i \in \mathbf{NR}(G) \setminus \mathcal{I}^t} \mathbb{E}_{\sigma_i^2 \mid G, \boldsymbol{x}^{1:t-1}}\left[\frac{N_t}{2}\log(2\pi\sigma_i^2 e)\right] + \mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}}\left[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}}\left[p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\right]\Bigg] \tag{C.51}$$

C.4 Derivation of the Causal Reasoning Utility Function

We derive the utility function $U_{\text{CR}}(a)$ in Eq. (4.12) for the query $Y = q_{\text{CR}}(\mathcal{M}) = p^{\text{do}(X_i=\psi)}(X_j \mid \mathcal{M})$ with $\psi \sim p(\psi)$ a distribution over intervention values. Starting with the joint entropy in Eq. (C.3) we marginalise over graphs (instead of SCMs) to exploit that we can sample from and evaluate $p(\boldsymbol{X} \mid G, \boldsymbol{x}^{1:t-1})$ in closed form by using GPs:

$$-H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})$$

$$= \mathbb{E}_{Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}}\left[\log p(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1})\right] \tag{C.52}$$

$$= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}}\left[\mathbb{E}_{Y, \boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}}\left[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}}\left[p(Y, \boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\right]\right] \tag{C.53}$$

$$= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}}\Big[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}}\Big[\mathbb{E}_{Y \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1}}\Big[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}}\left[p(Y \mid \boldsymbol{X}^t, G', \boldsymbol{x}^{1:t-1}) \cdot p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\Big]\Big]\Big] \tag{C.54}$$

Now, we need to clarify how to sample from and evaluate $p(Y \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1})$. This may seem problematic at first, as $q_{\text{CR}}(\mathcal{M})$ returns a set of density functions. What we want to learn, however, is not a set of explicitly represented densities, but rather a set of densities implicitly represented by a (posterior) SCM. Said another way, we want to learn (parts of) SCMs that induce the set of densities with high posterior $p(Y \mid \boldsymbol{x}^{1:t})$. Hence, to estimate $\mathbb{E}_{Y \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1}}[\cdot]$ we first sample intervention values $\psi \sim p(\psi)$ and then sample from the respective interventional densities $p^{\text{do}(X_i=\psi)}(X_j \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1})$ induced by candidate SCMs with graph G. Therefore, the expectation becomes $\mathbb{E}_\psi\left[\mathbb{E}^{\text{do}(X_i=\psi)}_{X_j \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1}}[\cdot]\right]$. To evaluate $p(Y \mid \boldsymbol{X}^t, G', \boldsymbol{x}^{1:t-1})$ we estimate $p^{\text{do}(X_i=\psi)}(X_j \mid \boldsymbol{X}^t, G', \boldsymbol{x}^{1:t-1})$ as described in Appx. D.1. The joint entropy therefore becomes

$$-H(Y, \boldsymbol{X}^t \mid \boldsymbol{x}^{1:t-1}) = \mathbb{E}_{G \mid \boldsymbol{x}^{1:t-1}}\Big[\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t-1}}\Big[\mathbb{E}_\psi\Big[\mathbb{E}^{\text{do}(X_i=\psi)}_{X_j \mid \boldsymbol{X}^t, G, \boldsymbol{x}^{1:t-1}}\Big[\log \mathbb{E}_{G' \mid \boldsymbol{x}^{1:t-1}}\left[p^{\text{do}(X_i=\psi)}(X_j \mid \boldsymbol{X}^t, G', \boldsymbol{x}^{1:t-1}) \cdot p(\boldsymbol{X}^t \mid G', \boldsymbol{x}^{1:t-1})\right]\Big]\Big]\Big]\Big] \tag{C.55}$$

By substituting Eqs. (C.13) and (D.1) into Eq. (C.12) we obtain the causal reasoning utility in Eq. (4.12).

D Approximate Inference and Experimental Details

In this section, we provide details about our approximate inference and estimation procedures, including the estimation of the marginal interventional likelihoods in Section D.1 and prior choices in Sections D.2 — D.4. We also provide details on DiBS for approximate graph posterior inference in Section D.5, the estimation of the information gain utilities in Section D.6, and our use of Bayesian Optimisation for experimental design in Section D.7.

D.1 Estimating Posterior Marginal Interventional Likelihoods

In the following, we show how we estimate (posterior) marginal interventional likelihoods $p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \boldsymbol{x}^{1:t})$. Let $\mathbf{Anc}_i^G$ and $\mathbf{Pa}_i^G$ denote the ancestor and parent sets of node X_i in G. Then, the marginal interventional likelihood is given by

$$
\begin{aligned}
&p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \boldsymbol{x}^{1:t}) \\
&= \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t}} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathcal{M}) \right] && \text{(D.1)} \\
&= \mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2, G \mid \boldsymbol{x}^{1:t}} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \boldsymbol{f}, \boldsymbol{\sigma}^2, G) \right] && \text{(D.2)} \\
&= \mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2, G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), \boldsymbol{f}, \boldsymbol{\sigma}^2, G} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{anc}_i^G, \boldsymbol{f}, \boldsymbol{\sigma}^2, G) \right] \right]. && \text{(D.3)}
\end{aligned}
$$

Given that X_i is independent of it's non-descendants given its parents, we obtain

$$
\begin{aligned}
&= \mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2, G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), \boldsymbol{f}, \boldsymbol{\sigma}^2, G} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, f_i, \sigma_i^2, G) \right] \right] && \text{(D.4)} \\
&= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\boldsymbol{f}, \boldsymbol{\sigma}^2 \mid G, \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), \boldsymbol{f}, \boldsymbol{\sigma}^2, G} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, f_i, \sigma_i^2, G) \right] \right] \right]. && \text{(D.5)}
\end{aligned}
$$

Given that $p(\boldsymbol{f}, \boldsymbol{\sigma}^2 \mid G, \boldsymbol{x}^{1:t})$ factorises and $\mathbf{Anc}_i^G$ are independent of mechanisms and noise variances $\boldsymbol{f}, \boldsymbol{\sigma}^2$ of the non-ancestors of X_i, we have

$$
\begin{aligned}
&= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \Bigg[\mathbb{E}_{\boldsymbol{f}_{\mathbf{Anc}_i^G}, \boldsymbol{\sigma}^2_{\mathbf{Anc}_i^G} \mid G, \boldsymbol{x}^{1:t}} \Bigg[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), \boldsymbol{f}_{\mathbf{Anc}_i^G}, \boldsymbol{\sigma}^2_{\mathbf{Anc}_i^G}, G} \Bigg[\\
&\qquad\qquad \mathbb{E}_{f_i, \sigma_i^2 \mid G, \boldsymbol{x}^{1:t}} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, f_i, \sigma_i^2, G) \right] \Bigg] \Bigg] \Bigg]. && \text{(D.6)}
\end{aligned}
$$

Finally, marginalising out the functions and noise variances, we obtain

$$
\begin{aligned}
&= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\boldsymbol{f}_{\mathbf{Anc}_i^G}, \boldsymbol{\sigma}^2_{\mathbf{Anc}_i^G} \mid G, \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), \boldsymbol{f}_{\mathbf{Anc}_i^G}, \boldsymbol{\sigma}^2_{\mathbf{Anc}_i^G}, G} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, G) \right] \right] \right] && \text{(D.7)} \\
&= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), G} \left[p^{\mathrm{do}(x_j)}(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, G) \right] \right] && \text{(D.8)} \\
&= \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\mathbb{E}_{\mathbf{Anc}_i^G \mid \mathrm{do}(x_j), G} \left[p(\boldsymbol{x}_i \mid \mathbf{pa}_i^G, G) \Big|_{X_j = x_j} \right] \right]. && \text{(D.9)}
\end{aligned}
$$

We use Monte Carlo estimation to approximate the outer expectation of this quantity according to Eq. (4.10). To approximate the inner expectation by performing ancestral sampling from the interventional density $p^{\mathrm{do}(x_j)}(\boldsymbol{X} \mid G)$, where we use 50 samples when estimating the U_{CR} utility in Equation Eq. (4.12) and 200 samples when estimating the metrics described in Appx. F.

D.2 Sampling Ground Truth Graphs

When generating ground truth SCMs for evaluation, we sample causal graphs according to two random graph models. First, we sample scale-free graphs using the preferential attachment process presented by Barabási and Albert [6]. We use the `networkx.generators.barabasi_albert_graph` implementation provided in the NetworkX [29] Python package and interpret the returned, undirected graph as a DAG by only considering the upper-triangular part of its adjacency matrix. Before permuting the node labels, we generate graphs with in-degree 2 for nodes $\{X_i\}_{i=3}^d$ whereas X_1 and X_2 are always

root nodes. In addition, we consider Erdös-Renyi random graphs [20], where edges are sampled independently with probability $p = \frac{4}{d-1}$. After sampling edges, we choose a random ordering and discard any edges that disobey this ordering to obtain a DAG. Our choice of p yields an expected degree of 2. Unlike Lorch et al. [45], we do not provide our model with any kind of prior information on the graph structure.

D.3 Normal-Inverse-Gamma Prior for Root Nodes

We use a conjugate normal-inverse-gamma (N-Γ^{-1}) prior

$$p(f_i, \sigma_i^2 \mid G) = \text{N-}\Gamma^{-1}(\mu_i, \lambda_i, \alpha_i^R, \beta_i^R) \tag{D.10}$$

as the joint prior over functions and noise parameters for root nodes in G (see Section 4 and Fig. 2). In our experiments, we use $\mu_i = 0$, $\lambda_i = 0.1$, $\alpha_i^R = 50$ and $\beta_i^R = 25$. When generating ground truth SCMs, we draw one sample for $(f_i^\star, \sigma_i^{2,\star})$ from this prior for all i and leave it fixed thereafter. Closed-form expressions for the (posterior) marginal likelihood can be found, e.g., in [51].

D.4 Gamma Priors for GP Hyperparameters of Non-Root Nodes

We model non-root node mechanisms with GPs (see Section 4.1), where each GP has a set of hyperparameters $(\boldsymbol{\kappa}_i, \sigma_i^2)$ where $\boldsymbol{\kappa}_i = (\kappa_i^l, \kappa_i^o)$ includes a length scale and output scale parameter, respectively, and where σ_i^2 denotes the variance of the Gaussian noise variable U_i. In our experiments, we use $p(\sigma_i^2 \mid G) = \text{Gamma}(\alpha = 50, \beta = 500)$, $p(\kappa_i^o \mid G) = \text{Gamma}(\alpha = 100, \beta = 10)$ and $p(\kappa_i^l \mid G) = \text{Gamma}(\alpha = 30 \cdot |\mathbf{Pa}_i^G|, \beta = 30)$, where $|\mathbf{Pa}_i^G|$ denotes the size of the parent set of X_i in G.

D.5 DiBS for Approximate Posterior Graph Inference

DiBS [45] introduces a probabilistic latent space representation for DAGs to allow for efficient posterior inference in continuous space. Specifically, given some latent particle $\boldsymbol{z} \in \mathbb{R}^{d \times d \times 2}$ we can define an edge-wise generative model

$$p(G \mid \boldsymbol{z}) = \prod_{i=1}^{d} \prod_{\substack{j=1 \\ j \neq i}}^{d} p(G_{i,j} \mid \boldsymbol{z}) \tag{D.11}$$

where $G_{i,j} \in \{0, 1\}$ indicates the absence/presence of an edge from X_i to X_j in G, and a prior distribution

$$p(\boldsymbol{Z}) \propto \exp(-\beta \, \mathbb{E}_{G \mid \boldsymbol{Z}} [h(G)]) \prod_{i,j,k} \mathcal{N}(z_{i,j,k} \mid 0, 1) \tag{D.12}$$

where $h(G)$ is a scoring function quantifying the "degree of cyclicity" of G. β is a temperature parameter weighting the influence of the expected cyclicity in the prior. Lorch et al. [45] propose to use Stein Variational Gradient Descent [44] for approximate inference of $p(\boldsymbol{Z} \mid \boldsymbol{x}^{1:t})$. SVGD maintains a fixed set of particles $\boldsymbol{z} = \{\boldsymbol{z}_m\}_{m=1}^M$ and updates them using the posterior score $\nabla \log p(\boldsymbol{z} \mid \boldsymbol{x}^{1:t}) = \nabla \log p(\boldsymbol{z}) + \nabla \log p(\boldsymbol{x}^{1:t} \mid \boldsymbol{z})$. In our experiments, we use $K = 5$ latent particles. For the estimation of expectations as in Eq. (4.10), we use $K = 40$ MC graph samples unless otherwise stated, hence, a total of $M \cdot K = 200$ graphs, and we use the DiBS+ particle weighting. In contrast to the original DiBS version, we do not use the annealing parameter α to force the mass of $p(G|\mathbf{z})$ onto a single graph during training. For further details on the method and its implementation, we refer to the original publication [45] and the provided code.

D.6 Estimation of the Information Gain Utility Functions

When estimating the information gain utilities (see § 4.2 and Appx. C), we keep the set of Monte Carlo samples from the SCM posterior $p(\mathcal{M} \mid \boldsymbol{x}^{1:t})$ fixed for all evaluations of the chosen utility during a given experiment design phase at time t, i.e., during the optimisation for all candidate intervention sets and intervention targets. In our experiments, for the U_{CD} and U_{CML} utilities we sample 5 and 30 graphs to approximate the outer and inner expectations w.r.t. the posterior graphs, respectively.

We sample 100 hypothetical experiment outcomes with given batch size from $p(\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t})$ to approximate the expectation $\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t}}[\cdot]$.

For the U_{CR} utility we sample 3 and 9 graphs to approximate the outer and inner expectations w.r.t. the posterior graphs, respectively. We sample 50 hypothetical experiment outcomes with given batch size from $p(\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t})$ to approximate expectations of the form $\mathbb{E}_{\boldsymbol{X}^t \mid G, \boldsymbol{x}^{1:t}}[\cdot]$. To approximate the expectations $\mathbb{E}_{\psi}\left[\mathbb{E}^{\text{do}(X_i=\psi)}_{X_j \mid \boldsymbol{X}^t, G, \mathcal{D}}[\cdot]\right]$ we sample 5 intervention values from $p(\psi)$ and draw 3 samples from $p^{\text{do}(X_i=\psi)}(X_j \mid \boldsymbol{X}^t, G, \mathcal{D})$ for each intervention value.

D.7 Bayesian Optimisation for Experimental Design

In order to find the optimal experiment $a_t^\star = (\mathcal{I}^\star, \boldsymbol{x}_{\mathcal{I}}^\star)$ at time t, we compute the optimal intervention value $\boldsymbol{x}_{\mathcal{I}}^\star \in \arg\max_{\boldsymbol{x}} U(\mathcal{I}, \boldsymbol{x})$ for each candidate intervention target set $\mathcal{I}$ (see Eq. (4.15)). As the evaluation of our proposed utility functions $U(a)$ is expensive, we require an efficient approach for finding optimal intervention values using as few function evaluations as possible. Following von Kügelgen et al. [79], we employ *Bayesian optimisation* (BO) [46, 47] for this task and model our uncertainty in $U(\mathcal{I}, \boldsymbol{x})$ given previous evaluations $\mathcal{D}_{BO} = \{(\boldsymbol{x}_l, U(\mathcal{I}, \boldsymbol{x}_l))\}_{l=1}^k$ with a GP. We select a new candidate solution according to the GP-UCB acquisition function [75],

$$\mathbf{x}_{k+1} = \arg\max_{\mathbf{x}} \mu_k(\mathbf{x}) + \gamma \sigma_k(\mathbf{x}) \,, \tag{D.13}$$

where $\mu_k(\mathbf{x})$ and $\sigma_k(\mathbf{x})$ correspond to the mean and standard deviation of the GP predictive distribution $p(U(\mathcal{I}, \boldsymbol{x}) \mid \mathcal{D}_{BO})$ (see Appx. B). We then evaluate $U(\mathcal{I}, \boldsymbol{x}_{k+1})$ at the selected $\boldsymbol{x}_{k+1}$ and repeat. The scalar factor γ trades off exploitation with exploration. In our experiments, we set $\gamma = 1$ and run the GP-UCB algorithm 8 times for each candidate set of intervention targets.

Algorithm 2: Particle Resampling

Input: set of latent particles $\boldsymbol{z} = \{\boldsymbol{z}_k\}_{k=1}^K$
Output: set of resampled latent particles $\tilde{\boldsymbol{z}} = \{\tilde{\boldsymbol{z}}_k\}_{k=1}^K$
$\tilde{\boldsymbol{z}} \leftarrow \varnothing$ ▷initialise set of resampled particles
$N_{max} \leftarrow \lceil \frac{K}{4} \rceil$ ▷max. number of particles to keep
$\{w_k\}_{k=1}^K \leftarrow \left\{ \frac{p(\boldsymbol{z}_k \mid \boldsymbol{x}^{1:t})\, \tilde{p}(\boldsymbol{z}_k)}{\sum_k p(\boldsymbol{z}_k \mid \boldsymbol{x}^{1:t})\, \tilde{p}(\boldsymbol{z}_k)} \right\}$ ▷compute particle weights
$n_{kept} \leftarrow 0$
for w_k **in** `sort_descending`$(\{w_k\}_{k=1}^K)$ **do**
  if $n_{kept} < N_{max}$ **and** $w_k > 0.01$ **then**
    $\tilde{\boldsymbol{z}} \leftarrow \tilde{\boldsymbol{z}} \cup \{\boldsymbol{z}_k\}$ $n_{kept} \leftarrow n_{kept} + 1$
  end
  else
    $\boldsymbol{z}_{new} \sim p(\boldsymbol{Z})$
    $\tilde{\boldsymbol{z}} \leftarrow \tilde{\boldsymbol{z}} \cup \{\boldsymbol{z}_{new}\}$
  end
end

E Implementation Details

In this section, we give details about our implementation, including our particle resampling procedure in Section E.1, the sharing and caching of priors in Section E.2, a discussion of the computational complexity of our implementation in Section E.3, and finally some information on our code framework and computing resources in Section E.4. Our implementation is available at https://www.github.com/chritoth/active-bayesian-causal-inference.

E.1 Particle Resampling

As described in Alg. 1, we resample latent particles $\boldsymbol{z} = \{\boldsymbol{z}_k\}_{k=1}^K$ according to a predefined schedule instead of sampling new particles from the particle prior $p(\boldsymbol{Z})$ after each epoch. Although sampling new particles would allow for higher diversity in the graph Monte Carlo samples and their respective mechanisms, it also entails a higher computational burden as the caching of mechanism marginal log-likelihoods is not as effective anymore. On the other hand, keeping a subset of the inferred particles is efficient, because once we have inferred a "good" particle $\boldsymbol{z}_k$ that supposedly has a high posterior density $p(\boldsymbol{z}_k \mid \boldsymbol{x}^{1:t})$ it would be wasteful to discard the particle only to infer a similar particle again. Empirically, we found that keeping particles depending on their unnormalized posterior densities according to Alg. 2 does not diminish inference quality while increasing computational efficiency. In our experiments, we chose the following resampling schedule:

$$r_t = \begin{cases} 1 & \text{if} \quad t \in \{1,2,3,4,5,6,9\} \\ 1 & \text{if} \quad t \bmod 5 = 0 \\ 0 & \text{otherwise.} \end{cases}$$

E.2 Shared Priors and Caching of Marginal Likelihoods

We share priors for mechanisms and noise $p(f_i, \sigma_i \mid G)$, as well as for GP hyperparameters $p(\boldsymbol{\kappa}_i \mid G)$, across all graphs G that induce the same parent set $\mathbf{Pa}_i^G$. Consequently, not only the posteriors $p(f_i, \sigma_i \mid G, \boldsymbol{x}^{1:t})$ and $p(\boldsymbol{\kappa}_i \mid G, \boldsymbol{x}^{1:t})$, but also the GP marginal likelihoods $p(\boldsymbol{x}_i^{1:t} \mid G)$ and GP predictive marginal likelihoods $p(\boldsymbol{x}_i^{t+1} \mid G, \boldsymbol{x}^{1:t})$ can be shared across graphs with identical parent sets for node X_i. By caching the values of the computed GP (posterior) marginal likelihoods, we substantially save on computational cost when computing expectations of the form $\mathbb{E}_{G \mid \boldsymbol{z}} \left[p(\boldsymbol{x}^{1:t} \mid G)\, \phi(G) \right]$ and $\mathbb{E}_{G \mid \boldsymbol{z}} \left[p(\boldsymbol{x}^{t+1} \mid G, \boldsymbol{x}^{1:t})\, \phi(G) \right]$ where $\phi(G)$ is some quantity depending the graph.

Specifically, consider that $p(\mathbf{x}^{1:t}|G) = \prod_i p(\mathbf{x}_i^{1:t}|G)$ factorizes into the GP marginal likelihoods of the individual mechanisms, so for d nodes in the graph and N samples in $\mathbf{x}^{1:t}$ (counted over all time steps) the complexity of computing $p(\mathbf{x}^{1:t}|G)$ is $O(d \cdot N^3)$ for a fixed set of GP hyperparameters (for

simplicity we ignore that not all d mechanisms are modelled by GPs as some are root nodes, so this is not a tight bound). Thus, in the worst case, estimating $p(\mathbf{x}^{1:t}|z) = \mathbb{E}_{G|z}[p(\mathbf{x}^{1:t}|G)]$ with K graph samples would yield a complexity of $O(K \cdot d \cdot N^3)$. By caching the marginal likelihoods as outlined above we can rewrite the complexity $O(K \cdot d \cdot N^3)$ as $O(L \cdot N^3)$ where $L \leq K \cdot d$ denotes the number of unique mechanisms entailed by the set of K graph samples. Although this does not reduce the worst case complexity it nevertheless greatly alleviates the computational demand in practice.

The benefit of caching becomes even more pronounced as $p(G \mid \boldsymbol{z})$ concentrates is mass on a small set of similar graphs as a result of the inference process. In particular, when updating the latent particles using SVGD we do not need to recompute $p(\boldsymbol{x}^{1:t} \mid G)$ after we have once before sampled G, which greatly speeds up the gradient estimation of the particle posterior.

E.3 Computational Complexity

There are two main phases in our algorithm (disregarding the computation of metrics for evaluation), (i) the inference phase where we (approximately) infer the posterior over SCMs $p(\mathcal{M}|\mathbf{x}^{1:t})$ after collecting new experimental data, and (ii) the experimental design phase.

The inference phase has worst-case complexity in $O(T_{SVGD} \cdot (T_{HP} \cdot M \cdot K \cdot d \cdot N^3 + M^2 \cdot d^2))$ where T_{SVGD} is the number of SVGD update steps, T_{HP} is the number of GP hyperparameter update steps, M is the number of latent z particles, K is the number of graph samples per latent z particle, d is the number of nodes in the network, and N is the number of collected experimental samples in $\mathbf{x}^{1:t}$. The computation of the GP marginal likelihood dominates the complexity of the inference phase. To improve scalability we make use of shared priors and caching. Additionally, we update the GP hyperparameters according to a predefined schedule instead of doing so after each performed experiment. In our experiments, both measures reduce the factor $T_{HP} \cdot M \cdot K \cdot d$ significantly. For example, running inference with 5 freshly initialized z particles with 40 graph samples each on a scale-free SCM with 20 nodes updates the hyperparameters of (2970, 964, 177) GPs during SVGD update steps (1, 5, 10), and of less than 15 GPs after 20 SVGD update steps. Compared to $M \cdot K \cdot d = 4000$ in this example, the benefit is evident.

In the experimental design phase we parallelize finding the optimal intervention value for each candidate target node, so the complexity is basically the number of Bayesian optimization (BO) steps times the complexity of the utility we want to optimize for. For a general query (cf. Eq. (4.11)) we have complexity in $O(T_{BO} \cdot M \cdot K_{outer} \cdot S \cdot Q \cdot M \cdot K_{inner} \cdot (O(p(y|\mathcal{M}) + d \cdot N^3)))$ where T_{BO} is the number of Bayesian optimization iterations, M is the number of latent z particles, K_{outer} is the number of graph samples in the outer SCM expectation in, S is the number of simulated experiments per SCM, Q is the number of simulated queries, K_{inner} is the number of graph samples in the inner SCM expectation, $O(\, p(y|\mathcal{M})\,)$ is the complexity of evaluating the query likelihood and $d \cdot N^3$ is the complexity of evaluating the GP predictive posteriors for the simulated experiments. For the causal discovery and model learning utilities the complexity reduces to $O(T_{BO} \cdot M \cdot K_{outer} \cdot S \cdot M \cdot K_{inner} \cdot d \cdot N^3)$.

In summary, the complexity of our ABCI implementation is dominated the experimental design phase from a high-level perspective. On a lower level, the cubic scaling of GP inference is the major computational issue that we alleviate by caching the (posterior) marginal log-likelihoods (see Appx. E.2 for details). However, in a small data regime where experimental data is costly to obtain, GPs are not a prohibitive element in our inference chain. Furthermore, GP scaling issues could be alleviated, e.g., by using sparse GP approximation or any other kind of scalable Bayesian mechanism model. Disregarding issues of GP scaling, the estimation of the information gain utilities is still costly, simply because it requires many levels of nested sampling and too few Monte-Carlo samples will yield too noisy, in the worst case unusable utility estimates. We believe that in follow-up work much can be gained in terms of scalability as well as performance by incorporating recent advances in nested Monte-Carlo/information gain estimation techniques(e.g., [8, 28, 62]).

Finally, consider that a single estimation of the causal discovery utility for an SCM with 20 nodes with $N = 500$ previously collected experimental samples takes approximately 2 minutes on an off-the-shelf laptop. Thus, for 10 BO iterations we can do the experimental design phase in 20 minutes (assuming we parallelize the utility optimization for each node). In a practical application scenario one might be very willing to invest hours or days for the design phase before conducting a costly experiment.

E.4 Implementation and Computing Resources

Our Python implementation uses the PyTorch [54], GPyTorch [24], CDT [41], SKLearn [58], NetworkX [29] and BoTorch [5] packages, which greatly eased our implementation efforts. All of our experiments were run on CPUs. We parallelise the experiment design by running the optimisation process for each candidate intervention set on a separate core.

F Evaluation Metrics

In this section, we provide details on the metrics used to evaluate our method in Section 5 and Appx. G. In our experiments, we use (nested) Monte Carlo estimators to approximate intractable expectations.

Kullback-Leibler Divergence. We evaluate the inferred posterior over queries given observed data, $p(Y \mid \boldsymbol{x}^{1:t})$, to the true query distribution $p(Y \mid \mathcal{M}^\star)$ using the Kullback-Leibler Divergence (KLD), i.e.,

$$\mathrm{KL}(p(Y \mid \mathcal{M}^\star) || \, p(Y \mid \boldsymbol{x}^{1:t})) = \mathbb{E}_{Y \mid \mathcal{M}^\star} \left[\log p(Y \mid \mathcal{M}^\star) - \log p(Y \mid \boldsymbol{x}^{1:t}) \right] \tag{F.1}$$

$$= \mathbb{E}_{Y \mid \mathcal{M}^\star} \left[\log p(Y \mid \mathcal{M}^\star) - \log \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t}} \left[p(Y \mid \mathcal{M}) \right] \right]. \tag{F.2}$$

Query KLD. For $Y = q_{\text{CR}}(\mathcal{M}) = \{p^{\text{do}(X_3=\psi)}(X_5 \mid \mathcal{M})\}$ with $\psi \sim p(\psi)$ we have

$$\text{Query KLD} = \mathbb{E}_\psi \left[\mathrm{KL}(p^{\text{do}(X_3=\psi)}(X_5 \mid \mathcal{M}^\star) || \, p^{\text{do}(X_3=\psi)}(X_5 \mid \boldsymbol{x}^{1:t})) \right] \tag{F.3}$$

$$= \mathbb{E}_\psi \left[\mathbb{E}_{X_5 \mid \text{do}(X_3=\psi), \mathcal{M}^\star} \left[\log p^{\text{do}(X_3=\psi)}(X_5 \mid \mathcal{M}^\star) - \log p^{\text{do}(X_3=\psi)}(X_5 \mid \boldsymbol{x}^{1:t}) \right] \right]. \tag{F.4}$$

To approximate the outer two expectations, we keep a fixed set of samples for each ground truth SCM to enhance comparability between different ABCI runs. For $p^{\text{do}(X_3=\psi)}(X_5 \mid \boldsymbol{x}^{1:t})$, we use the estimator described in Section D.1.

SCM KLD. For $Y = q_{\text{CML}}(\mathcal{M}) = \mathcal{M}$, we have

$$\text{SCM KLD} = \mathrm{KL}(p(\mathcal{M} \mid \mathcal{M}^\star) || \, p(\mathcal{M} \mid \boldsymbol{x}^{1:t})) \tag{F.5}$$

$$= \mathbb{E}_{\mathcal{M} \mid \mathcal{M}^\star} \left[\log p(\mathcal{M} \mid \mathcal{M}^\star) - \log p(\mathcal{M} \mid \boldsymbol{x}^{1:t}) \right] \tag{F.6}$$

$$= 0 - \log p(\mathcal{M}^\star \mid \boldsymbol{x}^{1:t}) \tag{F.7}$$

$$= -\log \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t}} \left[p(\mathcal{M}^\star \mid \mathcal{M}) \right] \tag{F.8}$$

$$= -\log \mathbb{E}_{G, \boldsymbol{f}, \boldsymbol{\sigma}^2 \mid \boldsymbol{x}^{1:t}} \left[p(G^\star, \boldsymbol{f}^\star, \boldsymbol{\sigma}^{2,\star} \mid G, \boldsymbol{f}, \boldsymbol{\sigma}^2) \right] \tag{F.9}$$

$$= -\log \mathbb{E}_{G, \boldsymbol{f}, \boldsymbol{\sigma}^2 \mid \boldsymbol{x}^{1:t}} \left[p(G^\star \mid G) \, p(\boldsymbol{f}^\star \mid \boldsymbol{f}) \, p(\boldsymbol{\sigma}^{2,\star} \mid \boldsymbol{\sigma}^2) \right]. \tag{F.10}$$

Now note that $p(G^\star \mid G) = 1$ if $G = G^\star$ and 0 otherwise, $p(\boldsymbol{f}^\star \mid \boldsymbol{f}) = \delta(\boldsymbol{f}^\star - \boldsymbol{f})$, and $p(\boldsymbol{\sigma}^{2,\star} \mid \boldsymbol{\sigma}^2) = \delta(\boldsymbol{\sigma}^{2,\star} - \boldsymbol{\sigma}^2)$. Hence, the SCM KLD vanishes iff the SCM posterior $p(\mathcal{M} \mid \boldsymbol{x}^{1:t})$ collapses onto the true SCM $\mathcal{M}^\star$, and is infinite otherwise.

Average Interventional KLD. Computing the KLD for $Y = q_{\text{CML}}(\mathcal{M}) = \mathcal{M}$ is not useful for evaluation, since it vanishes when the SCM posterior $p(\mathcal{M} \mid \boldsymbol{x}^{1:t})$ collapses onto the true SCM $\mathcal{M}^\star$ and is infinite otherwise. For this reason, we report the *average interventional KLD* as a proxy metric, which we define as

$$\text{Avg. I-KLD} = \frac{1}{d} \sum_{i=1}^{d} \mathbb{E}_\psi \left[\mathrm{KL}(p^{\text{do}(X_i=\psi)}(X \mid \mathcal{M}^\star) || \, p^{\text{do}(X_i=\psi)}(X \mid \boldsymbol{x}^{1:t})) \right] \tag{F.11}$$

$$= \frac{1}{d} \sum_{i=1}^{d} \mathbb{E}_\psi \left[\mathbb{E}_{X \mid \text{do}(X_i=\psi), \mathcal{M}^\star} \left[\log p^{\text{do}(X_i=\psi)}(X \mid \mathcal{M}^\star) - \log p^{\text{do}(X_i=\psi)}(X \mid \boldsymbol{x}^{1:t}) \right] \right] \tag{F.12}$$

$$= \frac{1}{d} \sum_{i=1}^{d} \mathbb{E}_\psi \Big[\mathbb{E}_{X \mid \text{do}(X_i=\psi), \mathcal{M}^\star} \Big[\log p^{\text{do}(X_i=\psi)}(X \mid \mathcal{M}^\star) - \log \mathbb{E}_{\mathcal{M} \mid \boldsymbol{x}^{1:t}} \left[p^{\text{do}(X_i=\psi)}(X \mid \mathcal{M}) \right] \Big] \Big]. \tag{F.13}$$

As with the Query KLD, we keep a fixed set of MC samples per ground truth SCM to approximate the two outer expectations to enhance comparability between different ABCI runs.

Expected Structural Hamming Distance. The Structural Hamming Distance (SHD)

$$\mathrm{SHD}(G, G^\star) = \left| \{(i,j) \in G : (i,j) \notin G^\star\} \right| + \left| \{(i,j) \in G^\star : (i,j) \notin G\} \right| \tag{F.14}$$

denotes the simple graph edit distance, i.e., it counts the number of edges (i, j) that are present in the prediction graph G and not present in the reference graph $G^\star$ and vice versa. We report the expected SHD w.r.t. our posterior over graphs as

$$\text{ESHD}(G, G^\star) = \mathbb{E}_{G \mid \boldsymbol{x}^{1:t}} \left[\text{SHD}(G, G^\star)\right] \tag{F.15}$$

AUPRC. Following previous work [19, 21, 45, 77], we report the *area under the precision recall curve* (AUPRC) by casting graph learning as a binary edge prediction problem given our inferred posterior edge probabilities $p(G_{i,j} \mid \boldsymbol{x}^{1:t})$. Refer to e.g. Murphy [52] for further information on this quantity.

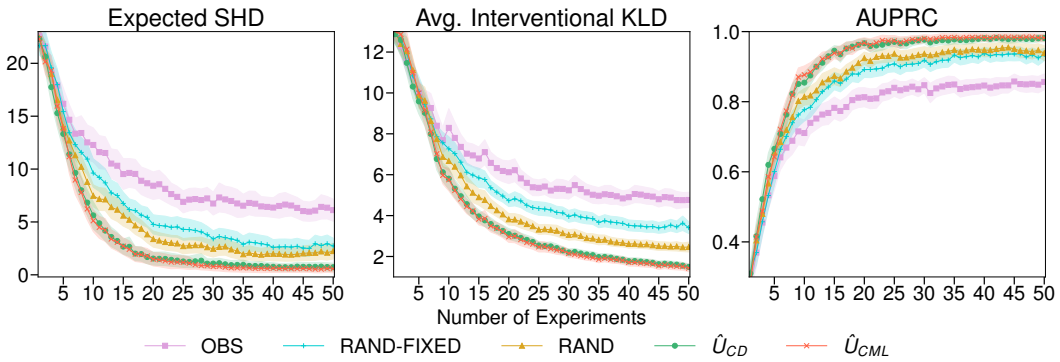

Figure 5: **Causal Discovery and SCM Learning on Scale-free Graphs with** 10 **Variables.** Comparison of the experimental design strategies with random and observational baselines on simulated ground truth models with 10 nodes. Lines and shaded areas show means and 95% confidence intervals (CIs) across 50 runs (10 randomly sampled ground-truth SCMs with 5 restarts per SCM). The U_{CD} and U_{CML} objectives perform on par with each other. Both clearly outperform the observational and random baselines on all metrics.

G Extended Experimental Results

Causal Discovery and SCM Learning for SCMs with $d = 10$ Variables. We report results on ground truth SCMs with $d = 10$ variables and scale-free graphs in Fig. 5. We initialise all methods with 5 observational samples and perform experiments with a batch size of 3. All other parameters are chosen as described in Appx. D.

Causal Discovery and SCM Learning for SCMs with $d = 20$ Variables. To demonstrate the scalability of our framework, we report results on ground truth SCMs with $d = 20$ variables and scale-free or Erdős-Renyi graphs in Fig. 6 and Fig. 7, respectively. We initialise all methods with 50 observational samples and perform experiments with a batch size of 5. All other parameters are chosen as described in Appx. D.

While ABCI shows clear benefits when scale-free causal graphs underlie the SCMs, we find that the advantage of ABCI diminishes on SCMs with unstructured Erdős-Renyi graphs, which appear to pose a harder graph identification problem. Moreover, we expect performance of our inference machinery, especially together with the informed action selection, to increase when investing more computational power to improve the quality of our estimates, e.g., by increasing the number of Monte Carlo samples used in our estimators and increasing the number of evaluations during the Bayesian optimisation phase.

Finally, in Fig. 8 we show that using a simple linear model (GP model with a linear kernel) is not able to reasonably capture the characteristics of the ground truth model (non-linear GP model) due to the model mismatch.

Learning Interventional Distributions vs. Causal Discovery and SCM Learning. We report additional metrics for our causal reasoning experiment as described in § 5 in Figs. 9 and 10. The key result here is that U_{CR} yields a significantly lower Query KLD while exhibiting a worse ESHD and Average I-KLD scores, which indicates that, indeed, the U_{CR} learns only those parts of the model that are relevant to reducing the uncertainty in our target query. This is more data efficient than trying to learn the entire model first and then answering the causal query of interest.

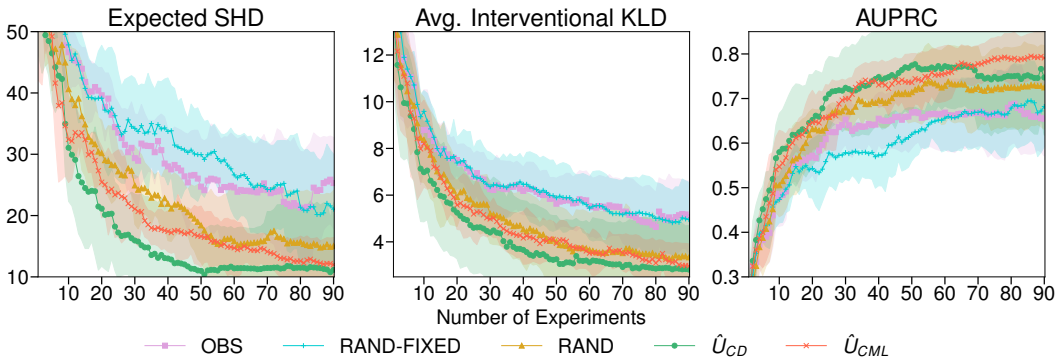

Figure 6: **Causal Discovery and SCM Learning on Scale-free Graphs with** 20 **Variables.** (Same figure as in Fig. 3 with additional confidence intervals for OBS and RAND FIXED.) Comparison of the experimental design strategy for causal discovery (U_{CD}) with random and observational baselines on simulated ground truth models with 20 nodes. Lines and shaded areas show means and 95% confidence intervals (CIs) across 15 runs (5 randomly sampled ground-truth SCMs with 3 restarts per SCM). The U_{CD} objective significantly outperforms the observational and random baselines on all metrics.

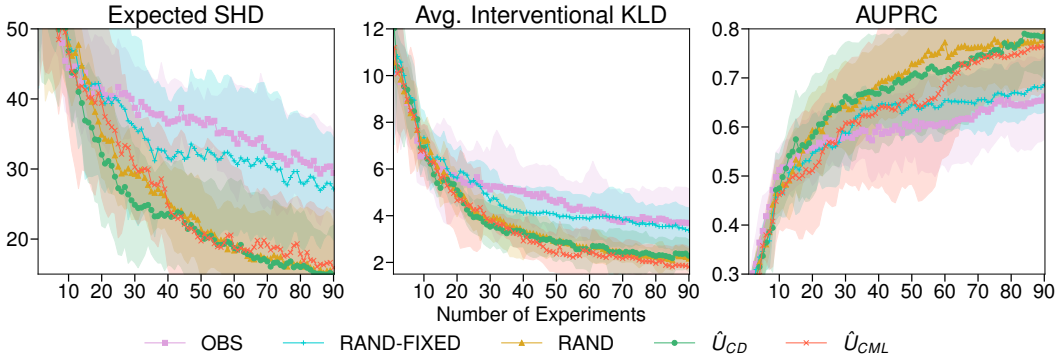

Figure 7: **Causal Discovery and SCM Learning on Erdős-Renyi Graphs with** 20 **Variables.** Comparison of experimental design strategies for causal discovery (U_{CD}) and causal model learning (U_{CML}) with random and observational baselines on simulated ground truth models with 20 nodes. Lines and shaded areas show means and 95% confidence intervals (CIs) across 15 runs (5 randomly sampled ground-truth SCMs with 3 restarts per SCM). The U_{CD} and U_{CML} strategies perform approx. equal to the strong random baseline (RAND) on all metrics, however, all three are significantly better than the weak random (RAND FIXED) and observational baselines. We expect that improving the quality of the U_{CD} and U_{CML} estimates (e.g., by scaling up computational resources invested in the MC estimates) yield similar benefits of the experimental design utilities as apparent in Fig. 6.

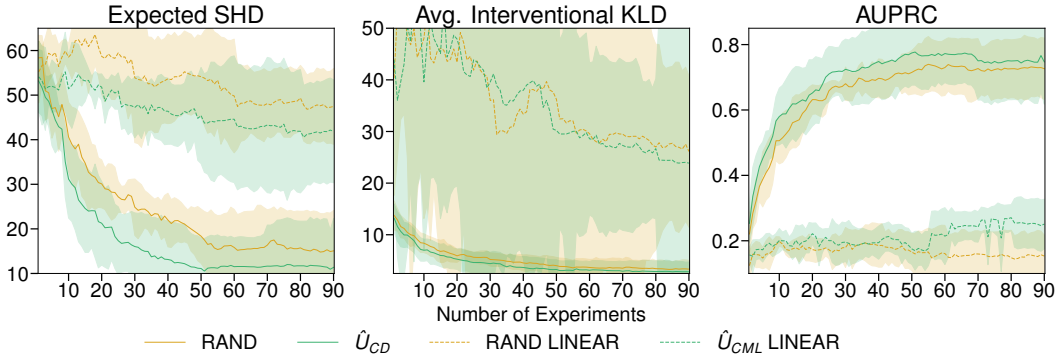

Figure 8: **Causal Discovery and SCM Learning on Scale-free Graphs with** 20 **Variables.** Comparison of non-linear GP model with a linear model (linear GP kernel) for U_{CD} an RAND on simulated ground truth models with 20 nodes. Lines and shaded areas show means and 95% confidence intervals (CIs) across 15 runs (5 randomly sampled ground-truth SCMs with 3 restarts per SCM). Clearly, the model mismatch in the linear model prohibits the identification of the ground-truth graph.

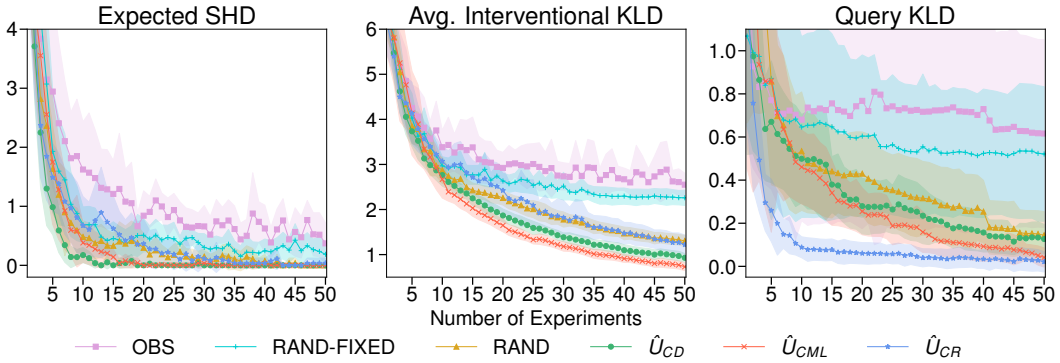

Figure 9: **Learning Interventional Distributions.** Comparison of the experimental design strategies with random and observational baselines. Lines and shaded areas show means and 95% confidence intervals (CIs) across 30 runs (10 randomly sampled ground-truth SCMs with 3 restarts per SCM). U_{CD}, U_{CML} and U_{CR} perform best w.r.t. the ESHD, Avg. I-KLD and Query KLD metrics respectively, which is expected.

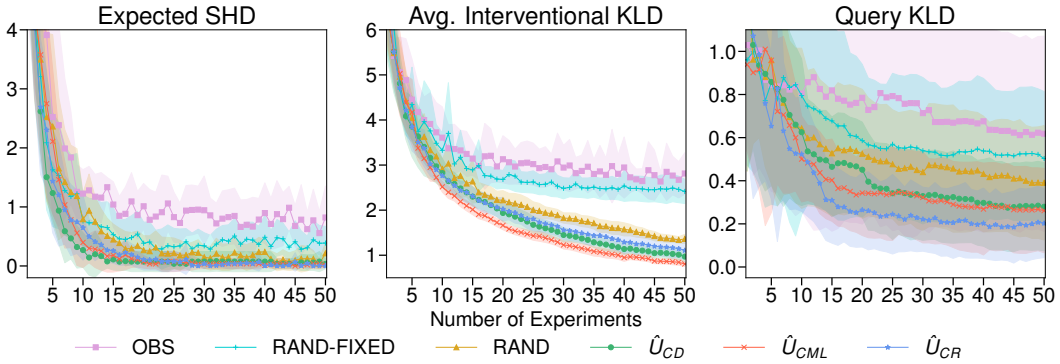

Figure 10: **Learning Interventional Distributions.** Comparison of the experimental design strategies with random and observational baselines. Lines and shaded areas show means and 95% confidence intervals (CIs) across 30 runs (10 randomly sampled ground-truth SCMs with 3 restarts per SCM). U_{CD}, U_{CML} and U_{CR} perform best w.r.t. the ESHD, Avg. I-KLD and Query KLD metrics respectively, which is expected.