Probabilistic and Causal Satisfiability: Constraining the Model

Markus Bläser
Saarland University, Saarbrücken, Germany

Julian Dörfler
Saarland University, Saarbrücken, Germany

Maciej Liśkiewicz
University of Lübeck, Germany

Benito van der Zander
University of Lübeck, Germany

Abstract

We study the complexity of satisfiability problems in probabilistic and causal reasoning. Given random variables $X_1, X_2, \ldots$ over finite domains, the basic terms are probabilities of propositional formulas over atomic events $X_i = x_i$, such as $\mathbb{P}(X_1 = x_1)$ or $\mathbb{P}(X_1 = x_1 \vee X_2 = x_2)$. The basic terms can be combined using addition (yielding linear terms) or multiplication (polynomial terms). The probabilistic satisfiability problem asks whether a joint probability distribution satisfies a Boolean combination of (in)equalities over such terms. Fagin et al. [11] showed that for basic and linear terms, this problem is NP-complete, making it no harder than Boolean satisfiability, while Mossé et al. [22] proved that for polynomial terms, it is complete for the existential theory of the reals.

Pearl's Causal Hierarchy (PCH) extends the probabilistic setting with interventional and counterfactual reasoning, enriching the expressiveness of the languages. However, Mossé et al. [22] found that the complexity of satisfiability remains unchanged. Van der Zander et al. [38] showed that introducing a marginalization operator to languages induces a significant increase in complexity.

We extend this line of work by adding two new dimensions to the problem by constraining the models. First, we fix the graph structure of the underlying structural causal model, motivated by settings like Pearl's do-calculus, and give a nearly complete landscape across different arithmetics and PCH levels. Second, we study small models. While earlier work showed that satisfiable instances admit polynomial-size models, this is no longer guaranteed with compact marginalization. We characterize the complexities of satisfiability under small-model constraints across different settings.

2012 ACM Subject Classification Theory of computation → Complexity theory and logic

Keywords and phrases Existential theory of the real numbers, Computational complexity, Probabilistic logic, Structural Causal Models

Digital Object Identifier 10.4230/LIPIcs.ICALP.2025.144

Category Track B: Automata, Logic, Semantics, and Theory of Programming

Related Version *Full Version*: `http://arxiv.org/abs/2504.19944`

Funding *Benito van der Zander*: Work supported by the Deutsche Forschungsgemeinschaft (DFG) grant 471183316 (ZA 1244/1-1).

1 Introduction

Reasoning about probability is essential in many research fields. In computer science, it plays a crucial role in analyzing probabilistic programs, understanding a program's behavior under probabilistic input assumptions, and handling uncertain information in expert systems. In a seminal paper, Fagin et al. [11], introduce a logic to reason about probabilities. Their aim was to formulate a calculus that allows to phrase statements like "$\mathbb{P}(X = x) = 1/2$" or

© Markus Bläser, Julian Dörfler, Maciej Liśkiewicz, and Benito van der Zander;
licensed under Creative Commons License CC-BY 4.0
52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025).
Editors: Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis
Article No. 144; pp. 144:1–144:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

"$\mathbb{P}(X = x) < \mathbb{P}(Y = y)$". They study (among other things) the satisfiability problem for Boolean combinations of such terms, i.e., is there a probability distribution that satisfies all the terms. It turns out that the expressiveness of the underlying calculus has a great influence on the complexity of such satisfiability problems. For instance, in their original work, Fagin et al. investigated three types of terms in the (in)equalities: basic ones, like the one above, which consist of only single basic probabilities, linear combinations of basic probabilities, and polynomial expressions in basic probabilities. This choice parameterizes the satisfiability problem and it turns out that the type of equations has an impact on the complexity of the corresponding satisfiability problem.

We will study the satisfiability of such formulas from a multi-parametric view: there are several parameters in the underlying calculus, like the type of equations mentioned above, and by adjusting these parameters, we get a large family of satisfiability of varying complexities, depending on the choice of our parameters. There has been a lot of work along these lines, see e.g. [11, 15, 22, 17, 38, 4, 16, 8], and the main contribution of this paper is that we finally (almost) complete the whole picture.

Another dimension is that we can enhance the basic probability terms. This is inspired by the causal theory in AI. The development of the modern causal theory in AI and empirical sciences has greatly benefited from an influential structured approach to inference about causal phenomena, which is based on a reasoning hierarchy named "Ladder of Causation", also often referred to as "Pearl's Causal Hierarchy" (PCH) ([36, 28, 3], see also [29] for a gentle introduction to the topic). This three-level framework formalizes various types of reasoning that reflect the progressive sophistication of human thought regarding causation. It arises from a collection of causal mechanisms that model the "ground truth" of *unobserved* nature formalized within a Structural Causal Model (SCM). These mechanisms are then combined with three patterns of reasoning concerning *observed* phenomena expressed at the corresponding layers of the hierarchy, known as *probabilistic* (also called *associational* in the AI literature), *interventional*, and *counterfactual* (for formal definitions of these concepts as well as for illustrative examples, see Section 2).

A basic term at the probabilistic (observational) layer is expressed as a common probability, such as[1] $\mathbb{P}(x, y)$. This may represent queries like "How likely does a patient have both diabetes ($X = x$) and high blood pressure ($Y = y$)?" This layer corresponds precisely to the calculus developed by Fagin et al. mentioned above. The interventional patterns extend the basic probability terms by allowing the use of Pearl's do-operator [28] which models an experiment like a randomized controlled trial [12]. For instance, $\mathbb{P}([x]y)$ which[2], in general differs from $\mathbb{P}(y|x)$, allows to ask hypothetical questions such as, e.g., "How likely it is that a patient's headache will be cured ($Y = y$) if he or she takes aspirin ($X = x$)?". An example formula at this layer is $\mathbb{P}([x]y) = \sum_z \mathbb{P}(y|x, z)\mathbb{P}(z)$ which estimates the causal effect of the intervention $do(X = x)$ (all patients take aspirin) on outcome variable $Y = y$ (headache cure). It illustrates the use of the prominent back-door adjustment to eliminate the confounding effect of a factor represented by variable Z [28]. The basic terms at the highest level of the hierarchy enable us to formulate queries related to counterfactual situations. For example,

[1] In our paper, we consider random variables over discrete, finite domains. By an event, we mean a propositional formula over atomic events of the form $X = x$, such as $(X = x \wedge Y = y)$ or $(X = x \vee Y \neq y)$. Moreover, by $\mathbb{P}(Y = y, X = x)$, etc., we mean, as usually, $\mathbb{P}(X = x \wedge Y = y)$. Finally by using lowercase letters in $\mathbb{P}(x, y)$, we abbreviate $\mathbb{P}(Y = y, X = x)$.

[2] A common and popular notation for the post-interventional probability is $\mathbb{P}(Y = y|do(X = x))$. In this paper, however, we use the notation $\mathbb{P}([X = x]Y = y)$ since it is more convenient for our analysis.

$\mathbb{P}([X = x]Y = y|(X = x', Y = y'))$ expresses the probability that, for instance, a patient who did not receive a vaccine ($X = x'$) and died ($Y = y'$) would have lived ($Y = y$) if he or she had been vaccinated ($X = x$).

The computational complexity aspects of reasoning about uncertainty in this framework have been the subject of intensive studies in the past decades. The research has resulted in a large number of significant achievements, especially in the case of probabilistic inference with the input probability distributions encoded by Bayesian networks [26]. These problems are of great interest in AI and important results from this perspective include [6, 7, 34, 25, 18]. However, despite intensive research, many fundamental problems in the field remain open.

The main interest of our research is focused on the precise characterization of the computational complexity of satisfiability problems (and their validity counterparts) for languages of all PCH layers, combined with increasing the expressiveness of (in)equalities by enabling the use of more complex operators. Our primary concern is the effect of constraining the model in these satisfiability problems, on the one hand by fixing the graph structure of the model, and on the other by bounding the model size to be polynomial.

1.1 Reasoning about probabilities: a multi-parametric view

One starting point for our investigations is a pioneering paper work by Fagin et al. [11], who introduce a logic to reason about probabilities. We are given a set of random variables over some fixed finite domain D, often $\{0, 1\}$. They consider formulas consisting of Boolean combinations of (in)equalities of *basic* and *linear* terms, like $\mathbb{P}((X = 0 \vee Y = 1) \wedge (X = 0 \vee Y = 0)) = 1 \wedge (\mathbb{P}(X = 0) = 0 \vee \mathbb{P}(X = 0) = 1) \wedge (\mathbb{P}(Y = 0) = 0 \vee \mathbb{P}(Y = 0) = 1)$ with binary variables X, Y. The authors provide a complete axiomatization for the used logic, which is essentially a formalization of Nilsson's probabilistic logic [24] and they show that the problem of deciding satisfiability is NP-complete. Thus, surprisingly, the complexity is no worse than that of propositional logic. Fagin et al. extend then the language to (in)equalities of *polynomial* terms, with the goal of reasoning about conditional probabilities. They prove that the latter variant is NP-hard and contained in PSPACE. Recently, Mossé et al. [22] have given the exact complexity of this problem, showing that deciding the satisfiability is $\exists\mathbb{R}$-complete, where $\exists\mathbb{R}$ is the well-studied class defined as the closure of the Existential Theory of the Reals (ETR) under polynomial time many-one reductions.

In this work, we study the satisfiability (and validity) problem for such formulas involving probabilities. As already seen above, the choice of the admissible operations have an impact on the complexity. There are further choices, which we can make and which we will discuss in the following. All these choices have an impact on the complexity. Therefore, we will phrase these satisfiability problems as multi-parametric or multi-dimensional problems. The first dimension is the expressiveness of the underlying arithmetic:

- **Arithmetic:** Basic, linear, or polynomial.

The second dimension will be the layer in the Pearl's Causal Hierarchy which we have already discussed in the previous section. Reasoning about uncertainty has been the subject of intensive studies in the past decades. The research has resulted in a large number of significant achievements, especially in the case of probabilistic inference with the input probability distributions encoded by Bayesian networks [26]. The setting is of great interest in AI since it allows one not only to reason about *observations*, but also about *causality*, and even *counterfactuals*, which is for instance important in fairness considerations. These three layers form a hierarchy of increasing expressiveness (for a more detailed description of the concepts, see Section 2).

- **PCH layer:** Probabilistic (observational), causal (interventional), or counterfactual.

The languages used in [11, 22] are capable of fully expressing probabilistic reasoning, in particular, they allow to express *marginalization*, which is a common paradigm in this field. However, in the frameworks discussed above, this ability is very limited since the languages *do not* allow the use of a unary summation operator Σ. Thus, for instance[3], to express the marginal distribution of a random variable Y over a subset of (binary) variables $\{Z_1, \ldots, Z_m\}$ as $\sum_{z_1,\ldots,z_m} \mathbb{P}(y, z_1, \ldots, z_m)$, an encoding without summation requires an expansion into $\mathbb{P}(y, Z_1 = 0, \ldots, Z_m = 0) + \ldots + \mathbb{P}(y, Z_1 = 1, \ldots, Z_m = 1)$, which of exponential size in m. Consequently, to analyze the complexity aspects of the studied problems, languages that exploit the standard notation of probability theory and statistics, one requires an encoding that represents marginalization via the sum operator Σ. In the recent paper [38], the authors present the first systematic study in this setting. They introduce a new natural class, named succ-$\exists\mathbb{R}$, which can be viewed as a succinct variant of $\exists\mathbb{R}$.

Given the impact of the compact marginalization operator, the way we can perform marginalization will be the third dimension:

- **Marginalization:** Expanded or compact.

For expanded marginalization, the complexity results by Fagin et al. [11], as well as Mossé [22] et al., can be easily adapted to capture all other parameter ranges, see the full version of our work. Therefore, throughout this paper, we only deal with the case when we have a compact marginalization operator in our calculus.

The fact that the decision problem for basic and linear arithmetic without summation is in NP is not obvious. A model for a formula is a joint probability distribution of n, say, binary variables, which is a table with 2^n entries. Furthermore, it is a priori not clear that the bit size of the entries is polynomial. It turns out, that these problems have what is called a *small model property.* If there is a model, that is, a joint probability distribution satisfying the expression, then there is one which has only polynomially many non-zero entries. This follows by linear algebra arguments. If the model is small, then it can be guessed efficiently in NP. In the case of polynomial arithmetic, we still have the small model property, but one cannot guarantee that the entries have polynomial size. However, the compact marginalization operator destroys the small model property that was crucial in the work of [11] and [22] and in fact increases the complexity of the satisfiability problems dramatically [38]. So the next dimension is the model size, which can either be polynomially bounded or arbitrary:

- **Model size:** Small (polynomially many entries) or unbounded.

The impact of restricting the model size in the presence of a compact marginalization operator will be one of the main topics in this work (see the results in Section 5).

Causal models are often represented as a graph or Bayesian network where all random variables are represented as nodes and the edges encode which variables influence each other. E.g., at the probabilistic level, the graph would encode which variables are conditionally independent of each other. In the satisfiability problems described so far, the graph was not mentioned, so the task was to decide whether there exists any model with any graph structure that satisfies the formula. In many studies, a researcher can infer information about

[3] We have chosen this example because it is short. In this particular example, the joint probability could be directly written as $\mathbb{P}(y)$ in the calculus of Fagin et al., see the formal definition in Section 2. This is however not true anymore if we sum over more complicated expressions.

the graph structure, combining observed and experimental data with existing knowledge. Indeed, learning graphical, causal structures from data has been the subject of a considerable amount of research (see, e.g., [21, 9, 13, 37] for recent reviews). Thus, we will study the satisfiability (resp. validity) of formulas with the additional requirement that the graph structure $\mathcal{G}$ is also given in the input. The goal would be to determine whether the formula is satisfied in a model (resp. is valid in all models) having the structure $\mathcal{G}$.

In fact, such a constraint validity problem is one of the central problems of causality. E.g., given a graph structure, Pearl's prominent do-calculus [28] is often applied to show the validity of the equivalence between causal and probabilistic expressions. The impact of including the graph in the input on the complexity will be the second important question that we will study in this work (see Section 4). Thus, as a fifth dimension, we study the model structure:

- **Model structure:** Specified by a graph as a part of the input or unconstrained.

1.2 A brief overview of our results

We continue by giving an informal description of our results. A detailed description can be found in Section 3.

We first show that constraining the graph structure of the model makes the problem only harder in the sense that the unconstrained problem can be reduced to the constrained one, since the unconstrained problem can always be reduced to the constrained one with the complete directed acyclic graph as a model. It turns out that for the probabilistic or interventional layer together with basic or linear arithmetic, there is indeed a jump. The probabilistic layer becomes at least $\exists\mathbb{R}$-hard, while the interventional layer jumps from PSPACE to NEXP. For all other combinations of the PCH layer and arithmetic, no such jump happens and the complexity stays the same, however, we require new proof techniques. The exact results are given compactly in Table 2 (for comparison, the complexity landscape for unconstrained models, can be found in Table 1).

Second, we consider the case when the underlying model is small; small here means that for the underlying probability distribution P, the number of tuples $(u_1, \ldots, u_m)$ with $P(U_1 = u_1, \ldots, U_m = u_m) > 0$ is polynomially bounded in the input size. For the interventional and counterfactual layer with polynomial arithmetic, this does not affect the complexity of the problem (again, new proof techniques are required), but for the probabilistic layer, the complexity drops down to $\exists\mathbb{R}^{\Sigma}$, a new class contained in PSPACE. The full results are given compactly in Table 3.

2 Preliminaries

2.1 Pearl's Causal Hierarchy: An example

To illustrate the main ideas behind Pearl's Causal Hierarchy (PCH), we present below an example that, we hope, will make it easier to understand the formal definitions given in Section 2.2. In this example, we consider a hypothetical scenario involving three attributes represented by binary random variables: Age, modeled by $Z = 1$ (young) and $Z = 0$ (old), (COVID-19) Vaccination represented by $X = 1$, if yes, and $X = 0$ if not vaccinated, and Recovery, represented by $Y = 1$ (and $Y = 0$ meaning mortality). Below, we describe a structural causal model (SCM) that represents an unobserved true mechanism underlying this scenario and illustrates the canonical patterns of reasoning expressible at different levels of the PCH.

U_1	U_2	U_3	$P(\mathbf{u})$	Z	X	Y
0	0	0	0.0474	0	1	0
0	0	1	0.4266	0	1	1
0	1	0	0.0126	0	0	1
0	1	1	0.1134	0	0	0
1	0	0	0.0316	1	0	1
1	0	1	0.2844	1	0	0
1	1	0	0.0084	1	1	1
1	1	1	0.0756	1	1	1

(a) Hidden

Z	X	Y	$P(z,x,y)$
0	0	0	0.1134
0	0	1	0.0126
0	1	0	0.0474
0	1	1	0.4266
1	0	0	0.2844
1	0	1	0.0316
1	1	1	0.0840

(b) Observed P

$P(\mathbf{u})$	Z	X	Y
0.0474	0	1	0
0.4266	0	1	1
0.0126	0	1	0
0.1134	0	1	1
0.0316	1	1	1
0.2844	1	1	1
0.0084	1	1	1
0.0756	1	1	1

(c) $do(X = 1)$

Z	Y	$P([X = 1]z, y)$
0	0	0.06
0	1	0.54
1	0	0.00
1	1	0.40

(d) Post-int. P

Figure 1 (a) *Unobserved true nature*: Probabilities of the unobserved variables and the induced by the mechanism $\mathcal{F}$ outcomes of the observed variables. (b) *Observational layer*: Probability distribution $P(z, x, y)$ of the observed variables. (c) *Interventional layer*: The modified mechanism $\mathcal{F}$ by using the operator $do(X = 1)$ (shown in (c)) leads to the *post-interventional* distribution (d), denoted as $P([X = 1]z, y)$, for $z, y \in \{0, 1\}$. A challenging task here is to compute the post-interventional distribution (d) from the observed distribution (b).

Structural Causal Model. An SCM is defined as a tuple $(\mathcal{F}, P, \mathbf{U}, \mathbf{X})$ which is of *unobserved nature* from the perspective of an empirical researcher. It specifies the distribution $P(\mathbf{U})$ of the population and the mechanism $\mathcal{F}$. In our example, the model assumes three independent binary random variables $\mathbf{U} = \{U_1, U_2, U_3\}$, with probabilities: $P(U_1 = 1) = 0.4, P(U_2 = 1) = 0.21, P(U_3 = 1) = 0.9$. They affect the observed endogenous (observed) random variables Z, X, Y via the mechanism $\mathcal{F} = \{F_1, F_2, F_3\}$ specified as follows: $Z := F_1(U_1) = U_1$; $X := F_2(Z, U_2) = ZU_2 + (1-Z)(1-U_2)$; $Y := F_3(Z, X, U_3) = XZ + (1-X)(1-U_3) + X(1-Z)U_3$.

Thus, the model determines the distribution $P(\mathbf{u})$, for $\mathbf{u} = (u_1, u_2, u_3)$, and the values for the observed variables, as can be seen in Fig. 1(a).

The unobserved random variable U_1 reflects the age of the population and Z is a function of U_1 (which may be more complex in a real population). Getting COVID-19 vaccination depends on age but also on other circumstances (severe allergic reaction to a component of the COVID-19 vaccine, moderate or severe acute illness, etc.) and this is modeled by U_2. So X is a function of Z and U_2. Finally, Y depends on age, being vaccinated, and on further circumstances like having other diseases, which are modeled by U_3. So Y is a function of Z, X, and U_3. The functions F_i define a directed graph structure on the random (observed) variables: there is an edge from A to B if B depends on A. We always assume that dependency graph of the SCM is acyclic. This property is also called *semi-Markovian*.

In our example, we get the DAG: $\begin{tikzcd} Z \arrow[r] \arrow[rr, bend left] & X \arrow[r] & Y \end{tikzcd}$.

Layer 1 (probabilistic). Empirical sciences primarily rely on observed data, typically represented as probability distributions over measurable variables. In our example, this corresponds to the distribution $P(Z, X, Y)$. The unobserved variables U_1, U_2, U_3 along with the causal mechanism $\mathcal{F}$, remain hidden from direct observation. A researcher thus receives the probabilities probabilities (shown in Fig. 1(b)) $P(z, x, y) = \sum_{\mathbf{u}} \delta_{\mathcal{F},\mathbf{u}}(z, x, y) \cdot P(\mathbf{u})$, where vectors $\mathbf{u} = (u_1, u_2, u_3) \in \{0, 1\}^3$ and $\delta_{\mathcal{F},\mathbf{u}}(z, x, y) = 1$ if $F_1(u_1) = z, F_2(z, u_2) = x$, and $F_3(z, x, u_3) = y$; otherwise $\delta_{\mathcal{F},\mathbf{u}}(z, x, y) = 0$. The relevant query in our scenario $P(Y = 1|X = 1)$ can be evaluated as $P(Y = 1|X = 1) = P(Y = 1, X = 1)/P(X = 1) = 0.5106/0.558 \approx 0.915$ which says that the probability for recovery ($Y = 1$) is high given that the patient was vaccinated ($X = 1$). On the other hand, the query for $X = 0$ can be evaluated as $P(Y = 1|X = 0) = 0.0442/0.442 = 0.1$, which may lead to the opinion that the vaccine is relevant to recovery. However, these results do not reflect a randomized controlled trial [12] that could establish the differences between a COVID-19 vaccination and a placebo vaccination.

Layer 2 (interventional). Consider a randomized trial in which each patient is vaccinated, denoted as $do(X = 1)$, regardless of age (Z) and other conditions (U_2). We model this by performing a hypothetical intervention in which we replace in $\mathcal{F}$ the mechanism $F_2(Z, U_2)$ by the constant function 1 and leaving the remaining functions unchanged (see, Fig. 1(c)). If $\mathcal{F}_{X=1} = \{F'_1 = F_1, F'_2 = 1, F'_3 = F_3\}$ denotes the new mechanism, then the *post-interventional* distribution $P([X = 1]Z, Y)$ is specified as $P([X = 1]z, y) = \sum_{\mathbf{u}} \delta_{\mathcal{F}_{X=1},\mathbf{u}}(z, y) \cdot P(\mathbf{u})$, where $\delta_{\mathcal{F}_{X=1},\mathbf{u}}$ denotes function δ as above, but for the new mechanism $\mathcal{F}_{X=1}$ (the distribution is shown in Fig. 1(d))[4]. To determine the causal effect of the COVID-19 vaccination on recovery, we compute, in an analogous way, the distribution $P([X = 0]Z, Y)$ after the intervention $do(X = 0)$, which means that all patients receive a placebo vaccination. Then, comparing the value $P([X = 1]Y = 1) = 0.95$ with $P([X = 0]Y = 1) = 0.0874$, we can conclude that $P([X = 1]Y = 1) - P([X = 0]Y = 1) > 0$. This can be interpreted as a positive (average) effect of the vaccine in the population. Note that it is not obvious how to compute the post-interventional distributions from the observed probability $P(Z, X, Y)$; Indeed, this is a challenging task in the field of causality.

Layer 3 (counterfactual). The key phenomena that can be modeled and analyzed at this level are counterfactual situations. Imagine, e.g., in our scenario, there is a group of patients who have not been vaccinated and died ($X = 0, Y = 0$). One may ask, what the outcome (Y) would be had they been vaccinated ($X = 1$). In particular, one can ask what the probability of recovery would be if we had vaccinated patients in this group. Using the formalism of Layer 3, we can express this as a counterfactual query: $P([X = 1]Y = 1|X = 0, Y = 0) = P([X = 1](Y = 1) \wedge (X = 0, Y = 0))/P(X = 0, Y = 0)$. Note that the event $[X = 1](Y = 1) \wedge (X = 0, Y = 0)$ incorporates simultaneously two counterfactual mechanisms: $\mathcal{F}_{X=1}$ and $\mathcal{F}$. This is the key difference to Layer 2, where we can only have one.

2.2 Syntax and semantics of probabilistic calculi

In this section, we describe the underlying formalism of the satisfiability problems that we consider. This will cover the first three parameter dimensions: the underlying arithmetic, with or without compact marginalization, and the PCH level. The other two dimensions then naturally follow since they are obtained by restricting the underlying model.

We always consider discrete distributions in the probabilistic and causal languages studied in this paper. We represent the values of the random variables as $Val = \{0, 1, \ldots, c - 1\}$ and denote by $\mathbf{X}$ the set of random variables used in a system. Here, we use bold letters for sets of variables or sets of values. By capital letters $X_1, X_2, \ldots$, we denote the individual variables and assume, w.l.o.g., that they all share the same domain Val. A value of X_i is often denoted by the corresponding lowercase letter x_i or a natural number. In this section, we describe the syntax and semantics of the languages starting with probabilistic ones and then we provide extensions to the causal systems.

By an *atomic* event, we mean an event of the form $X = x$, where X is a random variable and x is a value in the domain of X. The language $\mathcal{E}_{prop}$ of propositional formulas over atomic events is defined as the closure of such events under the Boolean operators $\wedge$ and $\neg$.

[4] A common and popular notation for the post-interventional probability is $P(Z, Y|do(X = 1))$. In this paper, we use the notation $P([X = 1]Z, Y)$ since it is more convenient for analyses involving counterfactuals.

To specify the syntax of interventional and counterfactual events, we define the intervention and extend the syntax of $\mathcal{E}_{prop}$ to $\mathcal{E}_{post\text{-}int}$ and $\mathcal{E}_{counterfact}$, respectively, using the following grammars:

$$\begin{array}{rcll} \mathcal{E}_{prop} \text{ is defined by} & \mathbf{p} & ::= & X = x \mid \neg\mathbf{p} \mid \mathbf{p} \wedge \mathbf{p} \\ \mathcal{E}_{int} \text{ is defined by} & \mathbf{i} & ::= & \top \mid X = x \mid \mathbf{i} \wedge \mathbf{i} \\ \mathcal{E}_{post\text{-}int} \text{ is defined by} & \mathbf{p_i} & ::= & [\,\mathbf{i}\,]\,\mathbf{p} \\ \mathcal{E}_{counterfact} \text{ is defined by} & \mathbf{c} & ::= & \mathbf{p_i} \mid \neg\mathbf{c} \mid \mathbf{c} \wedge \mathbf{c}. \end{array}$$

Note that since $\top$ means that no intervention has been applied, we can assume that $\mathcal{E}_{prop} \subseteq \mathcal{E}_{post\text{-}int}$.

The PCH consists of three languages $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$, each of which is based on terms of the form $\mathbb{P}(\delta)$. For the (observational or associational) language $\mathcal{L}_1$, we have $\delta \in \mathcal{E}_{prop}$, for the (interventional) language $\mathcal{L}_2$, we have $\delta \in \mathcal{E}_{post\text{-}int}$ and for the (counterfactual) language $\mathcal{L}_3$, $\delta \in \mathcal{E}_{counterfact}$. The expressive power and computational complexity properties of the languages depend largely on the operations that we are allowed to apply to the basic terms. Allowing gradually more complex operators, we describe the languages which are the subject of our studies below. We start with the description of the languages $\mathcal{T}_i^*$ of terms, with $i = 1, 2, 3$, using the following grammars[5]

$\mathcal{T}_i^{\text{base}}$	$\mathbf{t} ::= \mathbb{P}(\delta_i)$	$\mathcal{T}_i^{\text{base}\langle\Sigma\rangle}$	$\mathbf{t} ::= \mathbb{P}(\delta_i) \mid \sum_x \mathbf{t}$
$\mathcal{T}_i^{\text{lin}}$	$\mathbf{t} ::= \mathbb{P}(\delta_i) \mid \mathbf{t} + \mathbf{t}$	$\mathcal{T}_i^{\text{lin}\langle\Sigma\rangle}$	$\mathbf{t} ::= \mathbb{P}(\delta_i) \mid \mathbf{t} + \mathbf{t} \mid \sum_x \mathbf{t}$
$\mathcal{T}_i^{\text{poly}}$	$\mathbf{t} ::= \mathbb{P}(\delta_i) \mid \mathbf{t} + \mathbf{t} \mid -\mathbf{t} \mid \mathbf{t} \cdot \mathbf{t}$	$\mathcal{T}_i^{\text{poly}\langle\Sigma\rangle}$	$\mathbf{t} ::= \mathbb{P}(\delta_i) \mid \mathbf{t} + \mathbf{t} \mid -\mathbf{t} \mid \mathbf{t} \cdot \mathbf{t} \mid \sum_x \mathbf{t}$

where δ_1 are formulas in $\mathcal{E}_{prop}$, $\delta_2 \in \mathcal{E}_{post\text{-}int}$, $\delta_3 \in \mathcal{E}_{counterfact}$.

The probabilities of the form $\mathbb{P}(\delta_i)$ are called *primitives* or *basic terms*. In the summation operator $\sum_x$, we have a dummy variable x which ranges over all values $0, 1, \ldots, c-1$. The summation $\sum_x \mathbf{t}$ is a purely syntactical concept which represents the sum $\mathbf{t}[^0/_x] + \mathbf{t}[^1/_x] + \ldots + \mathbf{t}[^{c-1}/_x]$, whereby $\mathbf{t}[^v/_x]$, we mean the expression in which all occurrences of x are replaced with value v. For example, for $Val = \{0, 1\}$, the expression $\sum_x \mathbb{P}(Y = 1, X = x)$ semantically represents $\mathbb{P}(Y = 1, X = 0) + \mathbb{P}(Y = 1, X = 1)$.

We note that the dummy variable x is not a (random) variable in the usual sense and that its scope is defined in the standard way.

In the table above, the terms in $\mathcal{T}_i^{\text{base}}$ are just basic probabilities with the events given by the corresponding languages $\mathcal{E}_{prop}$, $\mathcal{E}_{post\text{-}int}$, or $\mathcal{E}_{counterfact}$. Next, we extend terms by being able to compute sums of probabilities and by adding the same term several times, we also allow for weighted sums with weights given in unary. Note that this is enough to state all our hardness results. All matching upper bounds also work when we allow for explicit weights given in binary. In the case of $\mathcal{T}_i^{\text{poly}}$, we are allowed to build polynomial terms in the primitives. On the right-hand side of the table, we have the same three kinds of terms, but to each of them, we add a marginalization operator as a building block.

The polynomial calculus $\mathcal{T}_i^{\text{poly}}$ was originally introduced by Fagin, Halpern, and Megiddo [11] (for $i = 1$) to be able to express conditional probabilities by clearing denominators. While this works for $\mathcal{T}_i^{\text{poly}}$, this does not work in the case of $\mathcal{T}_i^{\text{poly}\langle\Sigma\rangle}$, since clearing denominators with exponential sums creates expressions that are too large. But we could introduce basic terms of the form $\mathbb{P}(\delta_i|\delta)$ with $\delta \in \mathcal{E}_{prop}$ explicitly. All our hardness proofs work without conditional probabilities but all our matching upper bounds are still true with explicit conditional probabilities.

[5] In the given grammars, we omit the brackets for readability, but we assume that they can be used in a standard way.

Expressions like $\mathbb{P}(X=1)+\mathbb{P}(Y=2)\cdot\mathbb{P}(Y=3)$ are valid terms in $\mathcal{T}_1^{\text{poly}}$ and $\sum_z \mathbb{P}([X=0](Y=1,Z=z))$ and $\sum_z \mathbb{P}(([X=0]Y=1),Z=z)$ are valid terms in the language $\mathcal{T}_3^{\text{poly}\langle\Sigma\rangle}$, for example.

Now, let $Lab = \{\text{base}, \text{base}\langle\Sigma\rangle, \text{lin}, \text{lin}\langle\Sigma\rangle, \text{poly}, \text{poly}\langle\Sigma\rangle\}$ denote the labels of all variants of languages. Then for each $* \in Lab$ and $i = 1,2,3$, we define the languages $\mathcal{L}_i^*$ of Boolean combinations of inequalities in a standard way:

$$\mathcal{L}_i^* \text{ is defined by } \quad \mathbf{f} ::= \mathbf{t} \leq \mathbf{t}' \mid \neg\mathbf{f} \mid \mathbf{f}\wedge\mathbf{f}, \quad \text{where } \mathbf{t},\mathbf{t}' \text{ are terms in } \mathcal{T}_i^*.$$

Although the language and its operations can appear rather restricted, all the usual elements of probabilistic and causal formulas can be encoded. Namely, equality is encoded as greater-or-equal in both directions, e.g. $\mathbb{P}(x)=\mathbb{P}(y)$ means $\mathbb{P}(x)\geq\mathbb{P}(y)\wedge\mathbb{P}(y)\geq\mathbb{P}(x)$.

The number 0 can be encoded as an inconsistent probability, i.e., $\mathbb{P}(X=1\wedge X=2)$. In a language allowing addition and multiplication, any positive integer can be easily encoded from the fact $\mathbb{P}(\top)\equiv 1$, e.g. $4\equiv(1+1)(1+1)\equiv(\mathbb{P}(\top)+\mathbb{P}(\top))(\mathbb{P}(\top)+\mathbb{P}(\top))$. If a language does not allow multiplication, one can show that the encoding is still possible. Note that these encodings barely change the size of the expressions, so allowing or disallowing these additional operators does not affect any complexity results involving these expressions.

To define the semantics of the languages, we use a structural causal model (SCM) as in [28, Sec. 3.2]. An SCM is a tuple $\mathfrak{M}=(\mathcal{F},P,\mathbf{U},\mathbf{X})$, such that $\mathbf{V}=\mathbf{U}\cup\mathbf{X}$ is a set of variables partitioned into exogenous (unobserved) variables $\mathbf{U}=\{U_1,U_2,\ldots\}$ and endogenous variables $\mathbf{X}$. The tuple $\mathcal{F}=\{F_1,\ldots,F_n\}$ consists of functions such that function F_i calculates the value of variable X_i from the values $(\mathbf{x},\mathbf{u})$ of other variables in $\mathbf{V}$ as $F_i(\mathbf{pa}_i,\mathbf{u}_i)$ [6], where $\mathbf{Pa}_i\subseteq\mathbf{X}$ and $\mathbf{U}_i\subseteq\mathbf{U}$. P specifies a probability distribution of all exogenous variables $\mathbf{U}$. Since variables $\mathbf{X}$ depend deterministically on the exogenous variables via functions F_i, $\mathcal{F}$ and P obviously define the joint probability distribution of $\mathbf{X}$.

Throughout this paper, we assume that domains of endogenous variables $\mathbf{X}$ are discrete and finite. In this setting, exogenous variables $\mathbf{U}$ could take values in any domain, including infinite and continuous ones. A recent paper [39] shows, however, that any SCM over discrete endogenous variables is equivalent for evaluating post-interventional probabilities to an SCM where all exogenous variables are discrete with finite domains.

As a consequence, throughout this paper, we assume that domains of exogenous variables $\mathbf{U}$ are discrete and finite, too.

For any basic $\mathcal{E}_{int}$-formula $X_i=x_i$ (which, in our notation, means $do(X_i=x_i)$), we denote by $\mathcal{F}_{X_i=x_i}$ the function obtained from $\mathcal{F}$ by replacing F_i with the constant function $F_i(\mathbf{v}):=x_i$. We generalize this definition for all interventions specified by $\alpha\in\mathcal{E}_{int}$ in a natural way and denote as $\mathcal{F}_\alpha$ the resulting functions. For any $\varphi\in\mathcal{E}_{prop}$, we write $\mathcal{F},\mathbf{u}\models\varphi$ if φ is satisfied for values of $\mathbf{X}$ calculated from the values $\mathbf{u}$. For $\alpha\in\mathcal{E}_{int}$, we write $\mathcal{F},\mathbf{u}\models[\alpha]\varphi$ if $\mathcal{F}_\alpha,\mathbf{u}\models\varphi$. And for all $\psi,\psi_1,\psi_2\in\mathcal{E}_{counterfact}$, we write (i) $\mathcal{F},\mathbf{u}\models\neg\psi$ if $\mathcal{F},\mathbf{u}\not\models\psi$ and (ii) $\mathcal{F},\mathbf{u}\models\psi_1\wedge\psi_2$ if $\mathcal{F},\mathbf{u}\models\psi_1$ and $\mathcal{F},\mathbf{u}\models\psi_2$.

Finally, for $\psi\in\mathcal{E}_{counterfact}$, let $S_{\mathfrak{M}}(\psi)=\{\mathbf{u}\mid\mathcal{F},\mathbf{u}\models\psi\}$.

We define $[\![\mathbf{e}]\!]_{\mathfrak{M}}$, for some expression $\mathbf{e}$, recursively in a natural way, starting with basic terms as follows $[\![\mathbb{P}(\psi)]\!]_{\mathfrak{M}}=\sum_{\mathbf{u}\in S_{\mathfrak{M}}(\psi)}P(\mathbf{u})$ and, for $\delta\in\mathcal{E}_{prop}$, $[\![\mathbb{P}(\psi|\delta)]\!]_{\mathfrak{M}}=[\![\mathbb{P}(\psi\wedge\delta)]\!]_{\mathfrak{M}}/[\![\mathbb{P}(\delta)]\!]_{\mathfrak{M}}$, assuming that the expression is undefined if $[\![\mathbb{P}(\delta)]\!]_{\mathfrak{M}}=0$. For two expressions

[6] We consider recursive models, that is, we assume the endogenous variables are ordered such that variable X_i (i.e. function F_i) is not affected by any X_j with $j>i$. Here, we also use the usual notation with capital letters and lowercase letters where $\mathbf{Pa}_i$ are variables and $\mathbf{pa}_i$ is an assignment of values to those variables.

$\mathbf{e}_1$ and $\mathbf{e}_2$, we define $\mathfrak{M} \models \mathbf{e}_1 \leq \mathbf{e}_2$, if and only if, $[\![\mathbf{e}_1]\!]_{\mathfrak{M}} \leq [\![\mathbf{e}_2]\!]_{\mathfrak{M}}$. The semantics for negation and conjunction are defined in the usual way, giving the semantics for $\mathfrak{M} \models \varphi$ for any formula φ in $\mathcal{L}_3^*$.

2.3 Probabilistic and causal satisfiability problems

The (decision) satisfiability problems for languages of the PCH, denoted by $\textsc{Sat}^*_{\mathcal{L}_i}$, with $i = 1, 2, 3$ and $* \in \mathit{Lab}$, take as input a formula φ in $\mathcal{L}_i^*$ and ask whether there exists a model $\mathfrak{M}$ such that $\mathfrak{M} \models \varphi$. Analogously, the validity problems for $\mathcal{L}_i^*$ consist in deciding whether, for a given φ, $\mathfrak{M} \models \varphi$ holds for all models $\mathfrak{M}$. From the definitions, it is obvious that variants of the problems for the level i are at least as hard as their counterparts at the lower level.

In many studies, a researcher can infer information about the graph structure of the SCM, combining observed and experimental data with existing knowledge. Indeed, learning causal structures from data has been the subject of a considerable amount of research (see, e.g., [9, 13, 37] for recent reviews and [30, 23, 20, 33, 31] for current achievements). Thus, the problem of interest in this setting, which is a subject of this section, is to determine the satisfiability, resp., validity of formulas in SCMs with the additional requirement on the graph structure of the models.

Let $\mathfrak{M} = (\mathcal{F} = \{F_1, \ldots, F_n\}, P, \mathbf{U}, \mathbf{X} = \{X_1, \ldots, X_n\})$ be an SCM. We will assume that the models are *semi-Markovian* in the general case and *Markovian* in the graph constrained case[7], Markovian means that the exogenous arguments U_i, U_j of F_i, resp. F_j are independent whenever $i \neq j$. We define that a DAG $\mathcal{G} = (\mathbf{X}, E)$ represents the graph structure of $\mathfrak{M}$ if, for every X_j appearing as an argument of F_i, $X_j \to X_i$ is an edge in E. DAG $\mathcal{G}$ is called a *causal diagram* of the model $\mathfrak{M}$ [28, 3]. The satisfiability problems with an additional requirement on the causal diagram, take as input a formula φ and a DAG $\mathcal{G}$ with the task of deciding whether there exists an SCM with the structure $\mathcal{G}$ for φ. In this way, for problems $\textsc{Sat}^*_{\mathcal{L}_i}$, we get the corresponding problems denoted as $\textsc{Sat}^*_{\mathrm{DAG},\mathcal{L}_i}$, with $* \in \mathit{Lab}$.

An important feature of $\textsc{Sat}^{poly}_{\mathcal{L}_i}$ instances, over all levels $i = 1, 2, 3$, is the small model property which says that, for every satisfiable formula, there is a model whose size is bounded polynomially with respect to the length of the input. This property was used to prove the memberships of $\textsc{Sat}^{poly}_{\mathcal{L}_i}$ in NP and in $\exists\mathbb{R}$ [11, 15, 22, 16]. Interestingly, membership proofs of $\textsc{Sat}^{\mathrm{lin}\langle\Sigma\rangle}_{\mathcal{L}_1}$ in $\mathsf{NP}^{\mathsf{PP}}$ and $\textsc{Sat}^{\mathrm{lin}\langle\Sigma\rangle}_{\mathcal{L}_2}$ in PSPACE [8] rely on the property as well.

Apart from these advantages, the small model property is interesting in itself. For example, on the probabilistic layer, if a formula φ (without the summation) over variables $X_1, \ldots, X_n$ is satisfiable, then there exists a Bayesian network $\mathcal{B} = (\mathcal{G}, P_{\mathcal{B}})$, with a DAG $\mathcal{G}$ over nodes $X_1, \ldots, X_n$ and conditional probability tables $P_{\mathcal{B}}$ for each variable X_i, such that φ is true in $\mathcal{B}$ and the total size of the tables is bounded by a polynomial of $|\varphi|$.

These have motivated the introduction of the small-model problem, denoted as $\textsc{Sat}^{\mathrm{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$, which is defined like $\textsc{Sat}^{\mathrm{poly}\langle\Sigma\rangle}_{\mathcal{L}_1}$ with the additional constraint that a satisfying distribution should only have polynomially large support, that is, only polynomially many entries in the exponentially large table of probabilities are non-zero [4]. In the paper, the authors achieve

[7] We note that the general semi-Markovian model, which allows for the sharing of exogenous arguments and allows for arbitrary dependencies among the exogenous variables, can be reduced in a standard way to the Markovian model by introducing new auxiliary "latent" variables $L_1, \ldots, L_k$ which belong to the endogenous variables $\mathbf{X}$ of the model, but which are non-measurable in contrast to the "standard" observed / measurable variables. Then we can use such latent variables as nodes of a DAG but the joint probability distribution on $\mathbf{X}$ is restricted to only the measurable variables and ignores $L_1, \ldots, L_k$.

this by extending an instance with an additional unary input $p \in \mathbb{N}$ and requiring that the satisfying distribution has a support of size at most p. We define $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_2}$ and $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_3}$ in a similar way. Formally, we use the following:

▶ **Definition 1.** *The decision problems* $\text{SAT}^{poly\langle\Sigma\rangle}_{sm,\,\mathcal{L}_i}$*, with* $i = 1, 2, 3$*, take as input a formula* $\varphi \in \mathcal{L}_i^{poly\langle\Sigma\rangle}$ *and a unary encoded number* $p \in \mathbb{N}$ *and ask whether there exists a model* $\mathfrak{M} = (\mathcal{F}, P, \mathbf{U} = \{U_1, \ldots, U_m\}, \mathbf{X})$ *such that* $\mathfrak{M} \models \varphi$ *and* $\#\{(u_1, \ldots, u_m) : P(U_1 = u_1, \ldots, U_m = u_m) > 0\} \leq p$.

2.4 The (succinct) existential theory of the reals

For two computational problems A, B, we will write $A \leq_{\text{P}} B$ if A can be reduced to B in polynomial time, which means A is not harder to solve than B. A problem A is complete for a complexity class $\mathcal{C}$, if $A \in \mathcal{C}$ and, for every other problem $B \in \mathcal{C}$, it holds $B \leq_{\text{P}} A$. By co-$\mathcal{C}$, we denote the class of all problems A such that their complements $\overline{A}$ belong to $\mathcal{C}$.

To measure the computational complexity of $\text{SAT}^*_{\mathcal{L}_i}$, a central role play the following, well-known Boolean complexity classes NP, PSPACE, NEXP, and EXPSPACE (for formal definitions see, e.g., [2]). Recent research has shown that the precise complexity of several natural satisfiability problems can be expressed in terms of the classes over the real numbers $\exists\mathbb{R}$ and succ-$\exists\mathbb{R}$. Recall, that the existential theory of the reals (ETR) is the set of true sentences of the form

$$\exists x_1 \ldots \exists x_n \varphi(x_1, \ldots, x_n), \tag{1}$$

where φ is a quantifier-free Boolean formula over the basis $\{\vee, \wedge, \neg\}$ and a signature consisting of the constants 0 and 1, the functional symbols $+$ and $\cdot$, and the relational symbols $<$, $\leq$, and $=$. The sentence is interpreted over the real numbers in the standard way. The theory forms its own complexity class $\exists\mathbb{R}$ which is defined as the closure of the ETR under polynomial time many-one reductions [14, 5, 35]. A succinct variant of ETR, denoted as succ-ETR, and the corresponding class succ-$\exists\mathbb{R}$, have been introduced in [38]. succ-ETR is the set of all Boolean circuits C that encode a true sentence as in (1) as follows. Assume that C computes a function $\{0,1\}^N \to \{0,1\}^M$. Then $\{0,1\}^N$ represents the node set of the tree underlying φ and $C(i)$ is an encoding of the description of node i, consisting of the label of i, its parent, and its two children. The variables in φ are $x_1, \ldots, x_{2^N}$. As in the case of $\exists\mathbb{R}$, to succ-$\exists\mathbb{R}$ belong all languages which are polynomial time many-one reducible to succ-ETR. $\exists\mathbb{R}$ and succ-$\exists\mathbb{R}$ can also be defined in terms of real RAMs [10, 4]. These two classes correspond to nondeterministic polynomial and exponential time, respectively, on such machines.

In a breakthrough result, Renegar [32] showed that ETR $\in$ PSPACE. In addition, his result also shows that even if the formula contains an exponential number of arithmetic terms of exponential size with exponential degree, it is still possible to solve the formula in PSPACE, as long as the number of variables is polynomially bounded.

3 Results

Intuitively, one would expect that when we make the arithmetic of the underlying satisfiability problem more powerful, then the complexity of the problem will also increase. The same is true when we go up in the PCH. While this is sometimes the case, we will also see many interesting cases where the complexity remains unchanged when we change a certain parameter. Our main focus will be on the effect of constraining the model on the complexity

Table 1 The complexity landscape for *unconstrained models*. Sources: (a) for $\mathcal{L}_1$ [11], for $\mathcal{L}_2$ and $\mathcal{L}_3$ [22], (b) [22], (c) [8], (d) for $\mathcal{L}_1$ and $\mathcal{L}_2$ [38], for $\mathcal{L}_3$ [17, 8].

Terms	$\mathcal{L}_1$ (prob.)	$\mathcal{L}_2$ (interv.)	$\mathcal{L}_3$ (count.)
basic	NP (a)		
lin			
poly	$\exists\mathbb{R}$ (b)		
basic & marg.	$\mathsf{NP}^{\mathsf{PP}}$ (c)	PSPACE (c)	NEXP (c)
lin & marg.			
poly & marg.	succ-$\exists\mathbb{R}$ (d)		

Table 2 The complexity landscape for *constrained graph structures*. Sources: (a) Lemma 5, Proposition 2, and [8], (b) Theorem 7, (c) Proposition 2 and Theorem 11.

Terms	$\mathcal{L}_1$ (prob.)	$\mathcal{L}_2$ (interv.)	$\mathcal{L}_3$ (count.)
basic & marg.	$(\mathsf{NP}^{\mathsf{PP}} \cup \exists\mathbb{R})$-hard ($a$)	NEXP (b)	
lin & marg.			
poly & marg.	succ-$\exists\mathbb{R}$ (c)		

of the problem. As mentioned earlier, we will always assume a compact marginalization operator since this is a very natural operation that frequently appears in expressions involving probabilities. Furthermore, without compact marginalization, everything follows from the results [11, 22] since the models appearing in their proofs are already very constrained (see the full version).

3.1 Previous work

For unconstrained models, we have a clear picture of the satisfiability landscape, see Table 1. Fagin et al. [11] prove the NP-completeness for the probabilistic layer with basic and linear terms and the NP-hardness with polynomial terms. Mossé et al. [22] prove that the latter is in fact complete for the existential theory of the reals $\exists\mathbb{R}$. They also prove that the same complexity results also hold for the interventional and counterfactual layers of the PCH. This means that the layer of the PCH does not have any impact on the complexity of the corresponding satisfiability problem. Things however change, when we add a marginalization operator. As shown in [8] if the arithmetic is basic or linear, then the satisfiability problem is complete for $\mathsf{NP}^{\mathsf{PP}}$, PSPACE, and NEXP when the PCH layer increases from probabilistic to counterfactual. If, however, the arithmetic is polynomial, then all three layers have the same complexity again, they are complete for the new class succ-$\exists\mathbb{R}$ [38, 17, 8].

3.2 Our results

As our main result, we complete the complexity landscape of the probabilistic and causal satisfiability problems when we either constrain the graph structure of the SCM or when we force the model to be small.

As the first part of our main results, we classify the difficulty of the satisfiability problems when the graph structure is given as a part of the input. We will show that these cases are always at least as hard as the corresponding problems without specified graph structures.

Table 3 The complexity landscape for *satisfiability with small models*. Source: (a) Lemma 13 and [8], (b) Fact 3 and [4], and (c) Theorem 14 .

Terms	$\mathcal{L}_1$ (prob.)	$\mathcal{L}_2$ (interv.)	$\mathcal{L}_3$ (count.)
basic & marg.	$\mathsf{NP}^{\mathsf{PP}}$ (a)	PSPACE (a)	NEXP (a)
lin & marg.			
poly & marg.	$\exists\mathbb{R}^{\Sigma}$ (b)	NEXP (c)	

This is due to the fact that we can always take the complete acyclic graph. The most notable difference to the unconstrained problems happens at the interventional layer. Here we have an increase in complexity from PSPACE to NEXP. We also compare our results to the model checking problem. Here we are not only given a graph structure but the full SCM and we need to check whether the model satisfies the input formula. It turns out that model checking is considerably easier. For instance, when the arithmetic is polynomial with marginalization, then even at the probabilistic layer, the satisfiability problem is succ-$\exists\mathbb{R}$-complete, while the corresponding model checking problem is in $\mathsf{NP}^{\mathsf{PP}}$ (Theorem 4).

Secondly, we consider the satisfiability problems with the small model property and investigate how the complexity increases across the PCH, in the presence of summation operators. In Definition 1, we defined the small model property for SCMs. In the original work of Fagin et al., there was no underlying SCM since they only dealt with the probabilistic layer, and a small model meant that we only have polynomially many entries in the table of the joint probability distribution of the (observed) variables. Since each observed variable is a function in the unobserved variables, both definitions are equivalent for semi-Markovian models as we show in Fact 3. Most notably, with polynomial arithmetic, at the probabilistic layer, having a small model reduces the complexity to $\exists\mathbb{R}^{\Sigma}$, the existential theory of the reals enhanced with a compact summation operator. This was shown in [4] for a probabilistic model without an underlying SCM and with Fact 3 we show that it still holds for Definition 1. As soon as we are at the interventional or even counterfactual layer, then the complexity jumps, but only to NEXP. Thus when we have a small model, the problem is hard for a Boolean class, although we have polynomial arithmetic and would have expected that the problem is hard for some theory over the reals. For linear arithmetic with a compact summation operator, it is known that the complexity increases along the PCH, and we show that the corresponding proofs of [8] are still valid if one requires a small-model property. The results are summarized in Figure 3.

4 Satisfiability and validity with requirements on the graph structure of SCMs

4.1 Probabilistic layer

A Bayesian network (BN) $\mathcal{B} = (\mathcal{G}, P_{\mathcal{B}})$ consists of a DAG $\mathcal{G}$ over nodes representing random variables $X_1, \ldots, X_n$ and a distribution $P_{\mathcal{B}}$ which factorizes over $\mathcal{G}$, meaning that the joint probability $P_{\mathcal{B}}(x_1,\ldots,x_n)$ can be written as a product $\prod_{i=1}^{n} P_{\mathcal{B}}(x_i \mid \mathbf{pa}_i)$, where $\mathbf{pa}_i$ denotes the values $x_{j_1}, \ldots, x_{j_k}$ of parents $\mathbf{Pa}_i$ of X_i in $\mathcal{G}$. Then the distribution $P_{\mathcal{B}}$ is specified as a set of conditional probability tables for each variable X_i conditioned on its parents in $\mathcal{G}$ [18]. So, for example, in the case of binary variables, even if the array representing the joint

distribution has 2^n non-zero elements, it can still be compactly represented by a set of arrays of total size $\sum_{i=1}^{n} 2^{|\mathbf{Pa}_i|}$. This representation has many additional advantages, for example, enabling efficient probabilistic inference.

In the case of purely probabilistic languages, the problem $\textsc{Sat}^*_{\text{DAG},\mathcal{L}_1}$ can be read as follows: given a DAG $\mathcal{G}$ over nodes $\mathbf{X}$ of a Bayesian network (but not being given the distribution $P_\mathcal{B}$) and a formula φ in the language $\mathcal{L}_1^*$ decide whether there exists $P_\mathcal{B}$ such that φ is true in some BN $\mathcal{B} = (\mathcal{G}, P_\mathcal{B})$, where $P_\mathcal{B}$ factorizes over $\mathcal{G}$.

We start with the observation that $\textsc{Sat}^*_{\mathcal{L}_1}$ is not harder than $\textsc{Sat}^*_{\text{DAG},\mathcal{L}_1}$ for any $* \in \mathit{Lab}$.

▶ **Proposition 2.** *For all $* \in \mathit{Lab}$ and for any $\varphi \in \mathcal{L}_1^*$ over variables $X_1, \ldots, X_n$, it is true that $\varphi \in \textsc{Sat}^*_{\mathcal{L}_1}$ if and only if $(\varphi, \mathcal{G}_n) \in \textsc{Sat}^*_{\text{DAG},\mathcal{L}_1}$, where $\mathcal{G}_n$ denotes a complete DAG over nodes $X_1, \ldots, X_n$, with edges $X_i \to X_j$ for all $1 \le i < j \le n$.*

Proof. Assume φ is satisfiable in a BN $\mathcal{B} = (\mathcal{G}, P_\mathcal{B})$. Let $\mathcal{B}' = (\mathcal{G}_n, P_{\mathcal{B}'})$, where, for every i, we define the conditional probability table as $P_{\mathcal{B}'}(x_i \mid x_1,\ldots,x_{i-1}) := P_\mathcal{B}(x_i \mid x_1,\ldots,x_{i-1})$ if $P_\mathcal{B}(x_1,\ldots,x_{i-1}) > 0$; otherwise we set the value arbitrarily, e.g., $1/c$, where c is the cardinality of set of values of X_i. Due to the chain rule, we get that $P_{\mathcal{B}'}(x_1,\ldots,x_n) = P_\mathcal{B}(x_1,\ldots,x_n)$ which means that φ is satisfiable in $\mathcal{B}'$.

The reverse implication is obvious. ◀

Due to their great importance in probabilistic reasoning, a lot of attention in AI research has been given to both function and decision problems, in which the entire BN $\mathcal{B} = (\mathcal{G}, P_\mathcal{B})$ is given as input with the goal to answer specific queries to the distribution. Certainly, to the most basic ones belongs the BN-Pr problem to compute, for a given BN $\mathcal{B}$ over $\mathbf{X}$, a variable X in $\mathbf{X}$, and a value x, the probability $P_\mathcal{B}(X = x)$. The next important primitive of probabilistic reasoning consists in finding the Maximum a Posteriori Hypothesis (MAP). To study its computational complexity, the natural decision problem, named D-MAP, has been investigated, which asks if, for a given rational number τ, evidence $\mathbf{e}$, and some subset of variables $\mathbf{Q}$, there is an instantiation $\mathbf{q}$ to $\mathbf{Q}$ such that $P_\mathcal{B}(\mathbf{q}, \mathbf{e}) = \sum_\mathbf{y} P_\mathcal{B}(\mathbf{q}, \mathbf{y}, \mathbf{e}) > \tau$. It is well known that BN-Pr is #P-complete and D-MAP is $\mathsf{NP}^{\mathsf{PP}}$-complete [34, 25]. To relate our setting to the standard Bayesian reasoning, we introduce a Model Checking problem,

▶ **Definition 3.** *The Bayesian Network Model Checking problem, denoted as* BN-MC**, gets a BN $\mathcal{B} = (\mathcal{G}, P_\mathcal{B})$ and a formula φ in $\mathcal{L}_1^*$, and verifies whether the formula φ is satisfied in the BN $\mathcal{B}$.*

This problem can be considered as a generalization of the decision version of BN-Pr.

▶ **Theorem 4.** BN-MC$^{poly\langle\Sigma\rangle}$ *is in* $\mathsf{P}^{\#\mathsf{P}}$*. Equivalently, it is in* P^{PP}*.*

Combining this lemma with [8, Thm. 4], the result of [25], and Proposition 2, we get the following:

$$\text{BN-MC}^{\text{poly}\langle\Sigma\rangle} \le_{\text{P}} \text{D-MAP} \equiv_P \textsc{Sat}^{\text{base}\langle\Sigma\rangle}_{\mathcal{L}_1} \equiv_P \textsc{Sat}^{\text{lin}\langle\Sigma\rangle}_{\mathcal{L}_1} \le_{\text{P}} \textsc{Sat}^{\text{base}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1},$$

which presents the relationships between the model checking, D-MAP, and satisfiability problems for probabilistic languages from a complexity perspective. In the general case, i.e., in the case of polynomial languages with summation, the constraints on DAGs do not make the problems more difficult, but restricting the models to BNs leads to different complexities, under a standard complexity assumption.

▶ **Lemma 5.** $\textsc{Sat}^{base}_{\text{DAG},\mathcal{L}_1}$ *is* $\exists\mathbb{R}$*-hard.*

This means that $\text{SAT}^{\text{base}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1}$, which is at least as hard as either D-MAP or $\text{SAT}^{base}_{\text{DAG},\mathcal{L}_1}$, is both $\mathsf{NP}^{\mathsf{PP}}$-hard and $\exists\mathbb{R}$-hard. Since these are very distinct classes, one should not expect $\text{SAT}^{\text{base}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1}$ to be complete for either class.

▶ **Proposition 6.** $\text{SAT}^{poly\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1}$ *is* succ-$\exists\mathbb{R}$*-complete.*

Proof. From Theorem 11, we know that $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_3}$ is succ-$\exists\mathbb{R}$-complete. Since any probabilistic formula is a special case of a counterfactual one and $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_1}$ is succ-$\exists\mathbb{R}$-complete, we get $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1} \leq_{\text{P}} \text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_1}$. On the other hand, from Proposition 2, we can conclude that the opposite relation is also true. ◀

Thus $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_1} \equiv_P \text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1}$ and the problems are computationally harder than $\text{BN-MC}^{\text{poly}\langle\Sigma\rangle}$ if $\mathsf{P}^{\#\mathsf{P}} \subsetneq \mathsf{NEXP}$.

4.2 Interventional and counterfactual reasoning

One of the key components in the development of structural causal models is the do-calculus introduced by Pearl [27, 28], which allows one to estimate causal effects from observational data. The calculus can be expressed as a *validity* problem for intervention-level languages with the graph structure requirements on SCMs. In particular, the rules of the calculus can be seen as instances of validity problems for which equivalent graphical conditions in terms of d-separation are given. For example, the "insertion/deletion of observations" rule requires, for a given DAG $\mathcal{G}$, that, for all SCMs $\mathfrak{M}$ with the causal structure $\mathcal{G}$, and for all values $\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{w}$, it holds $\mathbb{P}([\mathbf{x}]\mathbf{y} \mid [\mathbf{x}](\mathbf{z}, \mathbf{w})) = \mathbb{P}([\mathbf{x}]\mathbf{y} \mid [\mathbf{x}]\mathbf{w})$. This rule can be written in a compact way as $\sum_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{w}} (\mathbb{P}([\mathbf{x}]\mathbf{y} \mid [\mathbf{x}](\mathbf{z}, \mathbf{w})) - \mathbb{P}([\mathbf{x}]\mathbf{y} \mid [\mathbf{x}]\mathbf{w}))^2 = 0$. In this section, we prove that, in general, the validity problem with polynomial arithmetic is very hard, namely we will see that it is complete for co-succ-$\exists\mathbb{R}$.

To this goal, we study the complexity of the satisfiabilities with graph structure requirements in the case of the interventional languages. The base and linear languages are neither weak enough to be independent of the graph structure nor strong enough to be able to distinguish graph structures exactly. In particular the complexity of $\text{SAT}^{\text{base}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ and $\text{SAT}^{\text{lin}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ increases above the PSPACE-completeness of $\text{SAT}^{\text{base}\langle\Sigma\rangle}_{\mathcal{L}_2}$ and $\text{SAT}^{\text{lin}\langle\Sigma\rangle}_{\mathcal{L}_2}$. On the other hand, the polynomial languages are strong enough to encode the DAG structure of the model which implies that the complexity for these languages is the same as $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_2}$.

▶ **Theorem 7.** $\text{SAT}^{base\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ *and* $\text{SAT}^{lin\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ *are* NEXP*-complete. Thus, the validity problems for the corresponding languages are complete for* co-NEXP*.*

An important tool for the proof of this theorem is the satisfiability of a Schönfinkel-Bernays sentence. The class of Schönfinkel–Bernays sentences (also called Effectively Propositional Logic, EPR) is a fragment of first-order logic formulas where satisfiability is decidable. Each sentence in the class is of the form $\exists\mathbf{x}\forall\mathbf{y}\psi$ whereby ψ can contain logical operations $\wedge, \vee, \neg$, variables $\mathbf{x}$ and $\mathbf{y}$, equalities, and relations $R_i(\mathbf{x}, \mathbf{y})$ which depend on a set of variables, but ψ cannot contain any quantifier or functions. Determining whether a Schönfinkel-Bernays sentence is satisfiable is an NEXP-complete problem [19] even if all variables are restricted to binary values [1].

▶ **Corollary 8.** $\text{SAT}^{base\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ *is computationally harder than* $\text{SAT}^{lin\langle\Sigma\rangle}_{\mathcal{L}_2}$ *unless* NEXP = PSPACE*.*

If the graph $\mathcal{G}$ is complete, that is, all nodes are connected by an edge, it is equivalent to a *causal ordering*, an ordering $X_{i_1} \prec X_{i_2} \prec \ldots \prec X_{i_n}$, such that variable X_{i_j} can only depend on variables X_{i_k} with $i_k < i_j$. If a causal ordering is given as a constraint, it does not change the complexity of the problem:

▶ **Lemma 9.** *In $\mathcal{L}_2$ one can encode a causal ordering.*

Proof. Given (in-)equalities in $\mathcal{L}_2$, we add a new variable C and add to each primitive the intervention $[C = 0]$. This does not change the satisfiability of (in-)equalities.

Given a causal order $V_{i_1} \prec V_{i_2} \prec \ldots$, we add c equations for each variable V_{i_j}, $j > 1$: $\mathbb{P}([C = 1, V_{i_{j-1}} = k]V_{i_j} = k) = 1$ for $k = 1, \ldots, c$.

The equations ensure that, if one variable is changed, and $C = 1$ is set, the next variable in the causal ordering has the same value, thus fixing an order from the first to the last variable. ◀

The equivalence between causal orderings and complete graphs thus results in:

▶ **Corollary 10.** $\text{SAT}^{lin\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ *remains* PSPACE*-complete in the special case that $\mathcal{G}$ happens to be a complete graph.*

Counterfactual languages on the other hand are strong enough to be able to enforce an explicit graph structure directly in the formula using exponential sums. That is, for each variable X_i, the formula can encode which variables are the parents of X_i by requiring that X_i remains constant if the parents remain constant. This condition is encoded with an intervention on the parents and an intervention on the other variables, and a counterfactual query ensuring that the second intervention does not change the probability distribution of X_i after the first intervention. An exponential sum can run this through all possible interventions.

▶ **Theorem 11.** $\text{SAT}^{*}_{\text{DAG},\mathcal{L}_3} \equiv_P \text{SAT}^{*}_{\mathcal{L}_3}$ *for any* $* \in \{base\langle\Sigma\rangle, lin\langle\Sigma\rangle, poly\langle\Sigma\rangle\}$.

Since we now know $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_1} \leq_{\text{P}} \text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2} \leq_{\text{P}} \text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_3} \equiv_P \text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_3}$, Proposition 6 together with $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_3}$ being succ-ETR complete [8], we obtain:

▶ **Corollary 12.** $\text{SAT}^{poly\langle\Sigma\rangle}_{\text{DAG},\mathcal{L}_2}$ *is* succ-*ETR complete.*

5 Probabilistic and causal reasoning with a small model

In [4], the complexity of a problem named $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ is investigated. The authors consider purely probabilistic models where *sm* means

$$\#\{(x_1, \ldots, x_m) : \mathbb{P}(X_1 = x_1, \ldots, X_n = x_m) > 0\}$$

is small, i.e., is polynomially bounded in the input size. Before talking further about $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$, we need to show that their problem $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ is equivalent to our problem $\text{SAT}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ according to Definition 1 where

$$\#\{(u_1, \ldots, u_m) : P(U_1 = u_1, \ldots, U_m = u_m) > 0\}$$

is small (we cannot use their definition, as it is not meaningful to give constraints on the distribution $\mathbb{P}(\mathbf{X})$ for *causal* models where interventions can change this distribution in various ways).

▶ **Fact 2.** *For a given model $\mathfrak{M}$, let $\#^+_{\mathbf{X}}(\mathfrak{M}) = \#\{(x_1, \ldots, x_n) : \mathbb{P}(X_1 = x_1, \ldots, X_n = x_n) > 0\}$ be the number of assignments to the observed variables with positive probability and $\#^+_{\mathbf{U}}(\mathfrak{M}) = \#\{(u_1, \ldots, u_m) : P(U_1 = u_1, \ldots, U_m = u_m) > 0\}$ be the number of assignments to the unobserved variables with positive probability.*

Then $\#^+_{\mathbf{X}}(\mathfrak{M}) \leq \#^+_{\mathbf{U}}(\mathfrak{M})$.

The reverse does not hold. All functions in the model might be constant, in which case $\#^+_{\mathbf{X}} = 1$, regardless of $\#^+_{\mathbf{U}}$. Thus a model that is a solution for the $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ problem of [4] might not be a solution to our problem $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$. Still:

▶ **Fact 3.** *For a given probabilistic formula φ and a number $p \in \mathbb{N}$, there exists a semi-Markovian model $\mathfrak{M} = (\mathcal{F}, P, \mathbf{U} = \{U_1, \ldots, U_m\}, \mathbf{X})$ such that $\mathfrak{M} \models \varphi$ and $\#^+_{\mathbf{U}}(\mathfrak{M}) \leq p$ if and only if there exists a model $\mathfrak{M}' = (\mathcal{F}', P', \mathbf{U}', \mathbf{X})$ such that $\mathfrak{M}' \models \varphi$ and $\#^+_{\mathbf{X}}(\mathfrak{M}') \leq p$.*

Thereby it is critical to consider semi-Markovian models, as the unobserved variables constructed in the proof might not be independent, as it was required in Markovian models.

▶ **Fact 4.** *Fact 3 does not hold for Markovian models.*

Nevertheless, Fact 3 implies that the $\exists\mathbb{R}^\Sigma$-completeness proof for a "$\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$" version restricted on the observed distribution in [4] also applies to our $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$. Thus we know that $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ is complete for a new complexity class $\exists\mathbb{R}^\Sigma$, a class that extends the existential theory of reals with summation operators. This is not a succinct class, so it is different from $\mathsf{succ}\text{-}\exists\mathbb{R}$, in fact they show

$$\exists\mathbb{R} \cup \mathsf{NP}^{\mathsf{PP}} \subseteq \exists\mathbb{R}^\Sigma \subseteq \mathsf{PSPACE} \subseteq \mathsf{NEXP} \subseteq \mathsf{succ}\text{-}\exists\mathbb{R} = \mathsf{NEXP}_{\text{real}},$$

where $\mathsf{NEXP}_{\text{real}}$ is the class of problems decided by nondeterministic real Random Access Machines (RAMs) in exponential time (see [4] for the exact details).

Not allowing multiplication reduces the complexity, however, once we allow for interventions, the complexity increases again. We know the following complexities for linear arithmetic from [8], whose results translate to small models:

▶ **Lemma 13.** *The following completeness results hold for small models:*

- $\textsc{Sat}^{basic\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ *and* $\textsc{Sat}^{lin\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$ *are* $\mathsf{NP}^{\mathsf{PP}}$*-complete.*
- $\textsc{Sat}^{basic\langle\Sigma\rangle}_{sm,\,\mathcal{L}_2}$ *and* $\textsc{Sat}^{lin\langle\Sigma\rangle}_{sm,\,\mathcal{L}_2}$ *are* PSPACE*-complete.*
- $\textsc{Sat}^{basic\langle\Sigma\rangle}_{sm,\,\mathcal{L}_3}$ *and* $\textsc{Sat}^{lin\langle\Sigma\rangle}_{sm,\,\mathcal{L}_3}$ *are* NEXP*-complete.*

Allowing multiplications again leads to NEXP-completeness, even on the second (interventional) level. Note however that this is likely still weaker than the $\mathsf{NEXP}_{\text{real}}$-completeness of $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_2}$ and $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{\mathcal{L}_3}$.

▶ **Theorem 14.** $\textsc{Sat}^{poly\langle\Sigma\rangle}_{sm,\,\mathcal{L}_2}$ *and* $\textsc{Sat}^{poly\langle\Sigma\rangle}_{sm,\,\mathcal{L}_3}$ *are* NEXP*-complete.*

This follows because a causal model consists of a probabilistic part $\mathbb{P}(\mathbf{u})$ and a deterministic, causal part $\mathcal{F}$. In the small model, $\mathbb{P}(\mathbf{u})$ is restricted to have polynomial size, but $\mathcal{F}$ can still have exponential size. In $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_1}$, one cannot reason about $\mathcal{F}$, but $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_2}$ and $\textsc{Sat}^{\text{poly}\langle\Sigma\rangle}_{sm,\,\mathcal{L}_3}$ can access the full power of $\mathcal{F}$. Since NEXP subsumes $\exists\mathbb{R}$, reasoning about polynomially many real numbers (probabilities) cannot increase the complexity.

6 Discussion

We investigated the computational complexities of probabilistic satisfiability problems from a multi-parametric point of view. Our new completeness results nicely extend and complement the previous achievements by [11, 22, 38, 4]. Our main focus was on the effect of constraining the model. We have shown that including a DAG in the input increases the complexity on the second level for linear languages (i.e., $\textsc{Sat}^{\mathrm{lin}\langle\Sigma\rangle}_{\mathrm{DAG},\mathcal{L}_2}$) to NEXP-complete, while it does not change the complexity on the third level or for the full languages. We leave the exact complexity of the first linear level (i.e., $\textsc{Sat}^{\mathrm{lin}\langle\Sigma\rangle}_{\mathrm{DAG},\mathcal{L}_1}$) for future work, but we conjecture that it is $\exists\mathbb{R}^{\Sigma}$-complete. On the other hand, including a Bayesian network, a DAG and its probability tables reduces the complexity to being in P^{PP}. Here, the completeness is left for future work[8].

Another interesting feature is that when the model is unconstrained, then the completeness for linear languages is expressed in terms of standard Boolean classes while the completeness of satisfiability for languages involving polynomials over the probabilities requires classes over the reals. However, once we constrain the model to be small, then the complexity with polynomial arithmetic is again described by a Boolean complexity class (for PCH layers two and three). This is due to the fact that the running time of Renegar's algorithm is essentially determined by the number of variables (which is small in the case of a small model) and not by the number of equations (which still can be large).

References

1. Antonis Achilleos. NEXP-completeness and universal hardness results for justification logic. In *International Computer Science Symposium in Russia*, pages 27–52. Springer, 2015. doi:10.1007/978-3-319-20297-6_3.
2. Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.
3. Elias Bareinboim, Juan D. Correa, Duligur Ibeling, and Thomas Icard. *On Pearl's Hierarchy and the Foundations of Causal Inference*, pages 507–556. Association for Computing Machinery, New York, NY, USA, 2022. doi:10.1145/3501714.3501743.
4. Markus Bläser, Julian Dörfler, Maciej Liśkiewicz, and Benito van der Zander. The existential theory of the reals with summation operators. In *35th Int. Symposium on Algorithms and Computation, ISAAC 2024*, volume 322 of *LIPIcs*, pages 13:1–13:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.ISAAC.2024.13.
5. John Canny. Some algebraic and geometric computations in PSPACE. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 460–467. ACM, 1988. doi:10.1145/62212.62257.
6. Gregory F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial intelligence*, 42(2-3):393–405, 1990. doi:10.1016/0004-3702(90)90060-D.
7. Paul Dagum and Michael Luby. Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, 60(1):141–153, 1993. doi:10.1016/0004-3702(93)90036-B.
8. Julian Dörfler, Benito van der Zander, Markus Bläser, and Maciej Liśkiewicz. From probability to counterfactuals: the increasing complexity of satisfiability in Pearl's Causal Hierarchy.

[8] As the main computation in the proof of Lemma 4 is performed by the PP-oracle; one can extend the proof to show the containment of BN-MC$^{\mathrm{poly}\langle\Sigma\rangle}$ in the space-limited complexity class L^{PP} and in the circuit complexity class $\mathsf{NC}_1^{\mathsf{PP}}$. Thus, it is unlikely to be P^{PP}-complete, although it might be possible that all three classes are identical.

In *International Conference on Learning Representations (ICLR)*. PMLR, to appear, 2025. Available as ArXiv TR 2405.07373.

9 Mathias Drton and Marloes H. Maathuis. Structure learning in graphical modeling. *Annual Review of Statistics and Its Application*, 4:365–393, 2017.

10 Jeff Erickson, Ivor Van Der Hoog, and Tillmann Miltzow. Smoothing the gap between NP and ER. *SIAM Journal on Computing*, 53:FOCS20–102–FOCS20–138, 2024.

11 Ronald Fagin, Joseph Y. Halpern, and Nimrod Megiddo. A logic for reasoning about probabilities. *Information and computation*, 87(1-2):78–128, 1990. doi:10.1016/0890-5401(90)90060-U.

12 Ronald Aylmer Fisher. Design of experiments. *British Medical Journal*, 1(3923):554, 1936.

13 Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

14 Dima Grigoriev and Nicolai Vorobjov. Solving systems of polynomial inequalities in subexponential time. *J. Symb. Comput.*, 5(1/2):37–64, 1988. doi:10.1016/S0747-7171(88)80005-1.

15 Duligur Ibeling and Thomas Icard. Probabilistic reasoning across the causal hierarchy. In *The 34th AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 10170–10177. AAAI Press, 2020. doi:10.1609/AAAI.V34I06.6577.

16 Duligur Ibeling, Thomas Icard, Krzysztof Mierzewski, and Milan Mossé. Probing the quantitative–qualitative divide in probabilistic reasoning. *Annals of Pure and Applied Logic*, 175(9):103339, 2024. doi:10.1016/J.APAL.2023.103339.

17 Duligur Ibeling, Thomas Icard, and Milan Mossé. On probabilistic and causal reasoning with summation operators. *Journal of Logic and Computation*, 2024.

18 Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

19 Harry R. Lewis. Complexity results for classes of quantificational formulas. *Journal of Computer and System Sciences*, 21(3):317–353, 1980. doi:10.1016/0022-0000(80)90027-6.

20 Ni Y. Lu, Kun Zhang, and Changhe Yuan. Improving causal discovery by optimal Bayesian network learning. In *Proc. of the AAAI Conference on Artificial Intelligence*, volume 35(10), pages 8741–8748, 2021. doi:10.1609/AAAI.V35I10.17059.

21 Nicolai Meinshausen, Alain Hauser, Joris M. Mooij, Jonas Peters, Philip Versteeg, and Peter Bühlmann. Methods for causal inference from gene perturbation experiments and validation. *Proceedings of the National Academy of Sciences*, 113(27):7361–7368, 2016.

22 Milan Mossé, Duligur Ibeling, and Thomas Icard. Is causal reasoning harder than probabilistic reasoning? *The Review of Symbolic Logic*, pages 1–26, 2022.

23 Ignavier Ng, Yujia Zheng, Jiji Zhang, and Kun Zhang. Reliable causal discovery with improved exact search and weaker assumptions. *Advances in Neural Information Processing Systems, NeurIPS*, 34:20308–20320, 2021. URL: https://proceedings.neurips.cc/paper/2021/hash/a9b4ec2eb4ab7b1b9c3392bb5388119d-Abstract.html.

24 Nils J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28(1):71–87, 1986. doi:10.1016/0004-3702(86)90031-7.

25 James D. Park and Adnan Darwiche. Complexity results and approximation strategies for MAP explanations. *Journal of Artificial Intelligence Research*, 21:101–133, 2004. doi:10.1613/JAIR.1236.

26 Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

27 Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

28 Judea Pearl. *Causality*. Cambridge University Press, 2009.

29 Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic books, 2018.

30 Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing*

Systems, NeurIPS, 34:27772–27784, 2021. URL: https://proceedings.neurips.cc/paper/2021/hash/e987eff4a7c7b7e580d659feb6f60c1a-Abstract.html.

31 Alexander Reisach, Myriam Tami, Christof Seiler, Antoine Chambaz, and Sebastian Weichwald. A scale-invariant sorting criterion to find a causal order in additive noise models. *Advances in Neural Information Processing Systems, NeurIPS*, 36:785–807, 2023.

32 James Renegar. On the computational complexity and geometry of the first-order theory of the reals. Part I: Introduction. Preliminaries. The geometry of semi-algebraic sets. The decision problem for the existential theory of the reals. *Journal of symbolic computation*, 13(3):255–299, 1992. doi:10.1016/S0747-7171(10)80003-3.

33 Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russell, Dominik Janzing, Bernhard Schölkopf, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *International Conference on Machine Learning, ICLM*, pages 18741–18753. PMLR, 2022. URL: https://proceedings.mlr.press/v162/rolland22a.html.

34 Dan Roth. On the hardness of approximate reasoning. *Artificial Intelligence*, 82(1-2):273–302, 1996. doi:10.1016/0004-3702(94)00092-1.

35 Marcus Schaefer. Complexity of some geometric and topological problems. In *International Symposium on Graph Drawing*, pages 334–344. Springer, 2009. doi:10.1007/978-3-642-11805-0_32.

36 Ilya Shpitser and Judea Pearl. Complete identification methods for the causal hierarchy. *Journal of Machine Learning Research*, 9(Sep):1941–1979, 2008. doi:10.5555/1390681.1442797.

37 Chandler Squires and Caroline Uhler. Causal structure learning: A combinatorial perspective. *Foundations of Computational Mathematics*, pages 1–35, 2022.

38 Benito van der Zander, Markus Bläser, and Maciej Liśkiewicz. The hardness of reasoning about probabilities and causality. In *Proc. Joint Conference on Artificial Intelligence (IJCAI 2023)*, 2023.

39 Junzhe Zhang, Jin Tian, and Elias Bareinboim. Partial counterfactual identification from observational and experimental data. In *International Conference on Machine Learning*, pages 26548–26558. PMLR, 2022. URL: https://proceedings.mlr.press/v162/zhang22ab.html.