Causal Modeling with Probabilistic Simulation Models*

Duligur Ibeling

Stanford University, Stanford, CA, USA
duligur@stanford.edu

Abstract. Recent authors have proposed analyzing conditional reasoning through a notion of intervention on a simulation program, and have found a sound and complete axiomatization of the logic of conditionals in this setting. Here we extend this setting to the case of probabilistic simulation models. We give a natural definition of probability on formulas of the conditional language, allowing for the expression of counterfactuals, and prove foundational results about this definition. We also find an axiomatization for reasoning about linear inequalities involving probabilities in this setting. We prove soundness, completeness, and NP-completeness of the satisfiability problem for this logic.

Keywords: Counterfactuals · conditional reasoning · probabilistic programs · conditional simulation.

1 Introduction

Accounts of subjunctive conditionals based on internal *causal models* offer an alternative to approaches based on ranking possible worlds by similarity [9]. One might, e.g., employ *structural equation models* (SEMs), i.e. systems of equations connecting the values of relevant variables, as the causal model; the semantics of conditionals are then based on a precise notion of *intervention* on the SEM [11]. Recently, some authors [8, 10, 4, 3, 1] have proposed using arbitrary programs, rather than systems of equations, as causal models. This approach emphasizes the procedural nature of many internal causal simulations over the purely declarative SEMs.

It is possible to define precisely this idea of programs as causal models and to generalize the idea of intervention from SEMs to programs [8]. It is also possible to give a sound and complete logic of conditionals in this setting [6]. However, these preliminary results have not fully explored the very important case—from, e.g., the Bayesian Logic modeling language [10] and implicit in the use of probabilistic programs as cognitive models [3]—of conditionals in a probabilistic setting, via using stochastic programs as the underlying causal model.

* Thanks to Thomas Icard for helpful discussions. The author was supported by the Sudhakar and Sumithra Ravi Family Graduate Fellowship in the School of Engineering at Stanford University for this work.

In the present contribution we will establish foundational definitions and logical results for this setting, thus extending the causal simulation framework to probabilistic simulation programs. Probabilities over a causal modeling language are defined and results showing that they may actually be interpreted as probabilities are given. The probabilities are used to give the semantics of a language for probabilistic reasoning, for which an axiomatization is given. The language and axiomatization are extensions of an analogous probabilistic language considered for the purely propositional case by [2]. Soundness and completeness of the axiom system is proven, and the satisfiability problem is found to be NP-complete.

2 Probabilistic Simulation Models and the Logical Language

2.1 Simulation Models

We work toward the definition of a language $\mathcal{L}$ for expressing probabilities involving probabilistic simulation models. Probabilistic simulation models extend the non-probabilistic[1] causal simulation models of [8, 6]. Formally, a *non-probabilistic simulation model* is a Turing machine[2], and a *probabilistic simulation model* is a probabilistic Turing machine, i.e., a deterministic Turing machine (that of course still has a read-write memory tape) given read access to a random bit tape whose squares represent the results of independent fair coin flips. The use of Turing machines is meant to allow for complete generality and encompasses, e.g., both logic programming and imperative programming. We sometimes use intuitive pseudocode in describing simulation models; such pseudocode is readily convertible to Turing machine code.

We suppose that simulation models are run initially from an empty tape.[3] As a simulation model runs, it reads and writes the values of binary variables on its tape squares. Eventually, the model either halts with some resultant tape, or does not halt, depending on the results of the coin flips the model performs in the course of its simulation. Every probabilistic simulation model thus induces a distribution on these possible outcomes. We are interested not only in these outcomes, but also in the dynamics and counterfactual information embodied in the model. That is, we are interested in what *would* happen were we to hold the

[1] The use of "non-probabilistic" rather than "deterministic" is intended to prevent confusion of the probabilistic/non-probabilistic distinction with the deterministic Turing machine/non-deterministic Turing machine distinction. The former distinction is about the presence of a source of randomness while the latter is about the number of possible halting executions.

[2] [6] does not require these machines to be deterministic, and isolates an additional logical principle that is valid when the machines are deterministic. However here we will suppose "non-probabilistic simulation model" always refers to one whose Turing machine is deterministic. This definition is more useful for comparison with the probabilistic case, in which all underlying machines are deterministic.

[3] [6] also includes an initial input tape in the definition of the model. This difference is inconsequential.

values of the tape square variables fixed in a particular way that counterfactually differs from the actual values the squares take on—in the distribution over outcomes that results under a particular *intervention*:

Definition 1 (Intervention [8]). *Let S be a specification of binary values for a finite number of tape squares: $S = \{x_i\}_{i \in I}$ for a finite index set $I \subseteq \mathbb{N}$. Then the intervention $\mathcal{I}_S$ is a computable function from Turing machines to Turing machines specified in the following way. Given a machine T, the intervened machine $\mathcal{I}_S(\mathsf{T})$ does the same thing as T but holds the variables in S to their fixed values specified by S throughout the run. That is, $\mathcal{I}_S(\mathsf{T})$ first writes x_i to square i for all $i \in I$, then runs T while ignoring any writes to any of the squares whose indices are in I.*

Suppose one fixes the entire random bit tape to some particular sequence in $\{0,1\}^\infty$. Then the counterfactual, as well as actual, behavior of a probabilistic simulation model is completely non-probabilistic. We define first a basic language that allows us to express facts about such behavior. Then we will define the probability that a given probabilistic simulation model satisfies a formula of this basic language. Our final language $\mathcal{L}$ uses these probabilities—it thus expresses facts about the *probabilities* that counterfactual properties hold. In all logical expressions we help ourselves to these standard notational conventions: $\alpha \to \beta$ abbreviates $\neg\alpha \vee \beta$, and $\alpha \leftrightarrow \beta$ denotes $(\alpha \to \beta) \wedge (\beta \to \alpha)$.

2.2 The Basic Language

Syntax The basic, non-probabilistic language $\mathcal{L}_{\text{non-prob}}$ is a propositional language over conditionals. Formally:

Definition 2. *Let X be a set of atoms $\{X_1, X_2, X_3, \dots\}$ representing the values of the memory tape variables and let $\mathcal{L}_{\text{prop}}$ be the propositional language formed by closing X off under conjunction, disjunction, and negation.*

Let the intervention specification language $\mathcal{L}_{\text{int}} \subset \mathcal{L}_{\text{prop}}$ be the language of purely conjunctive, ordered formulas of unique literals,[4] *i.e., formulas of the form $l_{i_1} \wedge \cdots \wedge l_{i_n}$ for some $n \geq 0$, where $i_j < i_{j+1}$ and each l_{i_j} is either X_{i_j} or $\neg X_{i_j}$. $\top$ abbreviates the "empty intervention" formula with $n = 0$. Let $\mathcal{L}_{\text{cond}}$ be the conditional language of formulas of the form $\langle\alpha\rangle\beta$ for $\alpha \in \mathcal{L}_{\text{int}}, \beta \in \mathcal{L}_{\text{prop}}$.*

The overall basic language $\mathcal{L}_{\text{non-prob}}$ is the language formed by closing off the formulas of $\mathcal{L}_{\text{cond}}$ [5] *under conjunction, disjunction, and negation.*

Every formula $\alpha \in \mathcal{L}_{\text{int}}$ specifies an intervention $\mathcal{I}_\alpha$ by giving a list of variables to fix and which values they are to be fixed to. Given a *subjunctive conditional*

[4] The point being that such formulas are in one-to-one correspondence with specifications of interventions, i.e., finite lists of variables along with the values each is to be held fixed to.

[5] Unlike [6], we do not admit the basic atoms X as atoms of $\mathcal{L}$. There is no difficulty extending the semantics to such atoms, but allowing them would needlessly complicate the proof of Theorem 1.

formula $\langle\alpha\rangle\beta \in \mathcal{L}_{\text{cond}}$, we call α the *antecedent* and β the *consequent*. We use $[\alpha]$ for the dual of $\langle\alpha\rangle$, i.e., $[\alpha]\beta$ abbreviates $\neg\langle\alpha\rangle(\neg\beta)$. Note that $\langle\rangle\varphi$ holds in a program if the unmodified program halts with a tape making φ true.

Semantics The semantics of the basic language are defined from considering a subjunctive conditional to be true in a simulation model when the program so intervened upon as to make its antecedent hold halts with such values of the tape variables as make its consequent hold. For example, consider a simple model that checks if the first memory tape square X_0 is 1 and if so writes a 1 into the second tape square X_1, and otherwise simply halts. This program satisfies the formulas $\langle\rangle\neg X_0$, $\langle\rangle\neg X_1$, but also the counterfactual formula $\langle X_0\rangle(X_0 \wedge X_1)$: holding the first memory square fixed to 1 causes a write of the value 1 into the second tape square, thus satisfying the consequent $X_0 \wedge X_1$. Formally:

Definition 3. *Let T be a non-probabilistic simulation model. Define $\mathsf{T} \models_{\text{non-prob}} \langle\alpha\rangle\beta$ iff $\mathcal{I}_\alpha(\mathsf{T})$ halts with a memory tape whose variable assignment satisfies β. Now suppose T is probabilistic, and fix values for all squares on the random bit tape to some sequence $\mathbf{r} \in \{0,1\}^\infty$. Define $\mathsf{T}, \mathbf{r} \models \langle\alpha\rangle\beta$ iff $\mathcal{I}_\alpha(\mathsf{T})$ when run with its random bit tape fixed to $\mathbf{r}$ halts with a resultant memory tape satisfying β. Define (in both cases) satisfaction of arbitrary formulas of $\mathcal{L}_{\text{non-prob}}$ in the familiar way by recursion.*

In a sense, the validities of the non-probabilistic setting carry over to this setting, as we will now show. For $\varphi \in \mathcal{L}_{\text{non-prob}}$, write $\models_{\text{non-prob}} \varphi$ if φ is valid in the class of all non-probabilistic simulation models. We will see that all such formulas are still valid for probabilistic simulation models, under Definition 3, once one fixes the random bit tape to a particular sequence.

Lemma 1. *$\models_{\text{non-prob}} \varphi$ if and only if, for all probabilistic simulation models T and all $\mathbf{r} \in \{0,1\}^\infty$, we have that $\mathsf{T}, \mathbf{r} \models \varphi$.*

Proof. Suppose $\models_{\text{non-prob}} \varphi$. Consider some probabilistic simulation model T and sequence $\mathbf{r} \in \{0,1\}^\infty$. φ is composed of $\mathcal{L}_{\text{cond}}$-atoms, of the form $\langle\alpha\rangle\beta$. What is the behavior of $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$? Either $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$ reads only a finite portion of $\mathbf{r}$ or reads an unbounded portion of $\mathbf{r}$ (in the latter case, it also does not halt). If only a finite portion is read, let $N(a)$ be the maximal random bit tape square reached of $\mathbf{r}$. Let N be the maximum of the $N(a)$ for all atoms a in φ, clearly existent as φ has finite length. Construct a Turing machine T' from T that embeds the contents of $\mathbf{r}$ up to index N into its code, replacing any read from $\mathbf{r}$ with its value. This is possible in a finite amount of code as we only have to include values up to N in T'.

What if $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$ ends up reading an unbounded portion of $\mathbf{r}$? We note that it is possible to write code in T' to check if the machine is being run under an α-fixing intervention—i.e., conditional code that runs under $\mathcal{I}_\alpha(\mathsf{T}')$ and no other intervention.[6] Add such code to T', including an infinite loop conditional on an

[6] For the precise details of this construction, see [6]. Briefly, if one wants to check if some X_i is being held fixed by an intervention, one can try to toggle X_i; this attempt will be successful iff X_i is not currently being fixed by an intervention.

α-intervention for each case where $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$ reads an unbounded portion of $\mathbf{r}$. Now, for all atoms $\langle\alpha\rangle\beta$, $\mathsf{T}' \models_{\text{non-prob}} \langle\alpha\rangle\beta$ iff $\mathsf{T}, \mathbf{r} \models \langle\alpha\rangle\beta$. As this holds for any atom of φ, and $\models_{\text{non-prob}} \varphi$, we have that $\mathsf{T}, \mathbf{r} \models \varphi$ as desired.

Now, suppose that $\mathsf{T}, \mathbf{r} \models \varphi$ for all probabilistic $\mathsf{T}, \mathbf{r}$. We want to see that $\models_{\text{non-prob}} \varphi$. Given a non-probabilistic T, convert T to a probabilistic TM T' that never reads from its random tape, and take any random tape $\mathbf{r}$. Then $\mathsf{T}', \mathbf{r} \models \varphi$ so that $\mathsf{T} \models_{\text{non-prob}} \varphi$. □

2.3 Adding Probabilities

Syntax $\mathcal{L}$ is the language of linear inequalities over probabilities that formulas of $\mathcal{L}_{\text{non-prob}}$ hold. More precisely:

Definition 4. *Let $\mathcal{L}_{\text{ineq}}$ be the language of formulas of the form*

$$a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c \tag{1}$$

for some $n \in \mathbb{N}$, and $c, a_1, \ldots, a_n \in \mathbb{Z}$, $\varphi_1, \ldots, \varphi_n \in \mathcal{L}_{\text{non-prob}}$. Then $\mathcal{L}$ is the language of propositional formulas formed by closing off $\mathcal{L}_{\text{ineq}}$ under conjunction, disjunction, and negation.

We sometimes write inequalities of a different form from (1) with the understanding that they can be readily converted into some $\mathcal{L}$-formula. For example, an inequality with a $>$ sign is a negation of a $\mathcal{L}_{\text{ineq}}$-formula.

Semantics Let T be a probabilistic simulation model. We will shortly define a probability $\mathbb{P}_\mathsf{T} : \mathcal{L}_{\text{non-prob}} \to [0, 1]$. Now suppose a given $\varphi \in \mathcal{L}_{\text{ineq}}$ has the form (1). Then $\mathsf{T} \models \varphi$ iff the inequality (1) holds when each $\mathbb{P}(\varphi_i)$ factor takes the value $\mathbb{P}_\mathsf{T}(\varphi_i)$. Satisfaction $\mathsf{T} \models \varphi$ for arbitrary $\varphi \in \mathcal{L}$ is then defined familiarly by recursion. Given $\varphi \in \mathcal{L}_{\text{non-prob}}$, the probability $\mathbb{P}_\mathsf{T}(\varphi)$ is simply the (standard) measure of the set of infinite bit sequences $\mathbf{r}$ for which $\mathsf{T}, \mathbf{r} \models \varphi$. More formally: let Σ be the σ-algebra on $\{0,1\}^\infty$ generated by cylinder sets and μ be the standard measure defined on Σ.[7] Now let $S(\varphi) = \{\mathbf{r} \in \{0,1\}^\infty : \mathsf{T}, \mathbf{r} \models \varphi\}$. Then we define $\mathbb{P}_T(\varphi) = \mu(S(\varphi))$. The following Lemma ensures that $S(\varphi)$ is always measurable, so that this definition is valid.

Lemma 2. *For any $\varphi \in \mathcal{L}_{\text{non-prob}}$, we have $S(\varphi) \in \Sigma$.*

Proof. Proof by induction on the structure of φ. If $\varphi = \neg\psi$, then $S(\varphi)$ is the complement of a set in Σ and hence is in Σ. The case of a conjunction or disjunction is similar since Σ is closed under intersection and union. The base case is that of the atoms. Consider an atom of the form $\langle\alpha\rangle\beta$. If $\mathcal{I}_\alpha(\mathsf{T})$ halts on $\mathbf{x}$ with random bit tape fixed to $\mathbf{r}$, then it does so reading only a finite portion of $\mathbf{r}$. Thus $S(\langle\alpha\rangle\beta)$ is the union of cylinder sets extending finite strings on which $\mathcal{I}_\alpha(\mathsf{T})$ halts with a result satisfying β, and hence is in Σ. □

[7] That is, as the product measure of Bernoulli(1/2) measures, as defined in, e.g., [3].

This probability is *coherent* in the sense that it plays well with the logic of the basic language:

Proposition 1. *For any probabilistic* T *we have,*

1. $\mathbb{P}_{\mathsf{T}}(\varphi) = 1$ *if* $\models_{\text{non-prob}} \varphi$ *for* $\varphi \in \mathcal{L}_{\text{non-prob}}$
2. $\mathbb{P}_{\mathsf{T}}(\varphi) \leq \mathbb{P}_{\mathsf{T}}(\psi)$ *whenever* $\models_{\text{non-prob}} \varphi \to \psi$ *for* $\varphi, \psi \in \mathcal{L}_{\text{non-prob}}$
3. $\mathbb{P}_{\mathsf{T}}(\varphi) = \mathbb{P}_{\mathsf{T}}(\varphi \wedge \psi) + \mathbb{P}_{\mathsf{T}}(\varphi \wedge \neg\psi)$ *for all* $\varphi, \psi \in \mathcal{L}_{\text{non-prob}}$

Proof. (1) holds since in this case, by Lemma 1, $S(\varphi) = \{0,1\}^\infty$. (2) holds since in this case, $S(\varphi) \subseteq S(\psi)$. Finally (3) holds by noting $\models_{\text{non-prob}} \varphi \leftrightarrow ((\varphi \wedge \psi) \vee (\varphi \wedge \neg\psi))$, applying (2), and noting that $S(\varphi \wedge \psi)$ and $S(\varphi \wedge \neg\psi)$ are disjoint. □

A corollary of part (2) is that logical equivalents under $\models_{\text{non-prob}}$ preserve probability.

2.4 The Case of Almost-Surely Halting Simulations

An interesting special case is that of the simulation models that halt almost-surely, i.e., with probability 1 under every intervention. Call this class $\mathcal{M}^{\downarrow}$. Following the urging of [7] we have not restricted the definition of probabilistic simulation model to such models. We will see that from a logical point of view, this case is a natural probabilistic analogue of the class $\mathcal{M}^{\downarrow}_{\text{non-prob}}$ of non-probabilistic simulation models that halt under every intervention. By this we mean that we may prove an analogue to Lemma 1. Write $\models^{\downarrow}_{\text{non-prob}} \varphi$ if $\varphi \in \mathcal{L}_{\text{non-prob}}$ is valid in $\mathcal{M}^{\downarrow}_{\text{non-prob}}$. Note that Lemma 1 does *not* hold if one merely changes all the preconditions to be halting/almost-surely halting: consider a probabilistic simulation model T that repeatedly reads random bits and halts at the first 1 it discovers; this program is almost-surely halting. But if $\mathbf{r}$ is an infinite sequence of 0s, then $\mathsf{T}, \mathbf{r} \not\models \langle\rangle\top$, even though $\models^{\downarrow}_{\text{non-prob}} \langle\rangle\top$. Crucially, we must move to the perspective of probability and measure to see the analogy:

Lemma 3. $\models^{\downarrow}_{\text{non-prob}} \varphi$ *if and only if, for all* $\mathsf{T} \in \mathcal{M}^{\downarrow}$*, we have* $\mathsf{T}, \mathbf{r} \models \varphi$ *for all* $\mathbf{r} \in \{0,1\}^\infty$ *except on a set of measure* 0*.*

Proof. Suppose $\models^{\downarrow}_{\text{non-prob}} \varphi$. We claim that for all $\mathsf{T} \in \mathcal{M}^{\downarrow}$ we have $\mathsf{T}, \mathbf{r} \models \varphi$ for all $\mathbf{r}$ except on a set of measure 0. Again, consider an atom $\langle\alpha\rangle\beta$ appearing in φ. The set of $\mathbf{r}$ for which $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$ does not halt has measure 0, given that $\mathsf{T} \in \mathcal{M}^{\downarrow}$. On each such $\mathbf{r}$, the run of $\mathcal{I}_\alpha(\mathsf{T}), \mathbf{r}$ must read infinitely many bits of $\mathbf{r}$: otherwise, the intervened machine would have a nonzero probability of not halting. Thus, excluding such $\mathbf{r}$, it is possible to repeat the construction of T' from the proof of Lemma 1 for $\langle\alpha\rangle\beta$, and in doing this construction we are already ignoring all cases where an unbounded portion of $\mathbf{r}$ is read. This means that we do not have to include any infinite loops in T', and T' will be always-halting. If we exclude all the such $\mathbf{r}$ arising from all antecedents of atoms of φ, then we only exclude a set of measure 0 since there are finitely many atoms. Except for such $\mathbf{r}$, the

construction works, and T' has, as before, the same behavior as T. But since $\models^{\downarrow}_{\text{non-prob}} \varphi$, we have that $\mathsf{T}, \mathbf{r} \models \varphi$ except on the excluded set of measure 0.

For the opposite direction, let $\mathsf{T} \in \mathcal{M}^{\downarrow}_{\text{non-prob}}$. We wish to show that $\mathsf{T} \models_{\text{non-prob}} \varphi$. Convert T to an identical probabilistic simulation program T' that never reads from its random tape. We have $\mathsf{T}', \mathbf{r} \models \varphi$ for all $\mathbf{r}$ but on a set of measure 0; in particular, for at least one $\mathbf{r}$. This implies $\mathsf{T} \models_{\text{non-prob}} \varphi$. □

3 Axiomatic Systems

We will now give an axiomatic system for reasoning in $\mathcal{L}$ and prove that it is *sound* and *complete* with respect to probabilistic simulation models: it proves all (completeness) and only (soundness) the formulas of $\mathcal{L}$ that hold for all probabilistic simulation models. We will give an additional system that is sound and complete for validities with respect to the almost-surely halting simulation models $\mathcal{M}^{\downarrow}$.

Definition 5. *Let AX be a set of rules and axioms formed by combining the following three modules.*

1. *PC: propositional reasoning (tautologies and* modus ponens*) over atoms of* $\mathcal{L}$.
2. *Prob: the following axioms:*

$$\begin{aligned}
\textit{NonNeg.}\quad & \mathbb{P}(\varphi) \geq 0 \\
\textit{Norm.}\quad & \mathbb{P}(\top) = 1 \\
\textit{Add.}\quad & \mathbb{P}(\varphi \wedge \psi) + \mathbb{P}(\varphi \wedge \neg\psi) = \mathbb{P}(\varphi) \\
\textit{Dist.}\quad & \mathbb{P}(\varphi) = \mathbb{P}(\psi) \text{ whenever } \models_{\text{non-prob}} \varphi \leftrightarrow \psi
\end{aligned}$$

3. *Ineq, an axiomatization (see [2]) for reasoning about linear inequalities:*

$$\begin{aligned}
\textit{Zero.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \\
& \Leftrightarrow (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) + 0\mathbb{P}(\varphi_{n+1}) \leq c) \\
\textit{Permutation.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \Leftrightarrow (a_{j_1}\mathbb{P}(\varphi_{j_1}) + \cdots + a_{j_n}\mathbb{P}(\varphi_{j_n}) \leq c) \\
& \text{when } j_1, \ldots, j_n \text{ are a permutation of } 1, \ldots, n \\
\textit{AddIneq.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \wedge (a'_1\mathbb{P}(\varphi_1) + \cdots + a'_n\mathbb{P}(\varphi_n) \leq c') \\
& \Rightarrow ((a_1 + a'_1)\mathbb{P}(\varphi_1) + \cdots + (a_n + a'_n)\mathbb{P}(\varphi_n) \leq (c + c')) \\
\textit{Mult.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \\
& \Rightarrow (ba_1\mathbb{P}(\varphi_1) + \cdots + ba_n\mathbb{P}(\varphi_n) \leq bc) \text{ for any } b > 0 \\
\textit{Dichotomy.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \vee (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \geq c) \\
\textit{Mono.}\quad & (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c) \\
& \Rightarrow (a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) < b) \text{ if } b > c
\end{aligned}$$

Additionally, let $\mathsf{AX}^{\downarrow}$ *be the system formed in exactly the same way, but replacing* $\models_{\text{non-prob}}$ *with* $\models^{\downarrow}_{\text{non-prob}}$.

Note that the non-probabilistic validities $\models_{\text{non-prob}}$ and $\models^{\downarrow}_{\text{non-prob}}$, appearing in Dist, have been completely axiomatized in [6]. The main result is:

Theorem 1. *AX (respectively, $\mathsf{AX}^{\downarrow}$) is sound and complete for the validities of $\mathcal{L}$ with respect to $\mathcal{M}$ (respectively, $\mathcal{M}^{\downarrow}$).*

Proof. Soundness (of Prob) follows from Lemma 1, Proposition 1, and, for the almost-surely halting case, Lemma 3. For completeness, consider the general case of $\mathcal{M}$ first. As usual, it suffices to show that any consistent $\varphi \in \mathcal{L}$ is satisfiable by some probabilistic simulation model. We put φ into a normal form from which we construct a canonical model. By PC we may suppose φ is in disjunctive normal form. We may further suppose that it is a conjunction of $\mathcal{L}_{\text{ineq}}$-literals, as at least one (conjunctive) clause in the disjunctive normal form must be consistent. Let $a_1, \ldots, a_n \in \mathcal{L}_{\text{cond}}$ be the atoms that appear inside any probability $\mathbb{P}$ in φ, and let $\delta_1, \ldots, \delta_{2^n}$ represent all the formulas of the form $l_1 \wedge \cdots \wedge l_n$ that can be obtained by setting each l_i to either a_i or $\neg a_i$. We then have the following, which is a kind of normal form result:

Lemma 4 (Lemma 2.3, [2]). *φ is provably-in-AX equivalent to a conjunction*

$$\begin{aligned}
&(\mathbb{P}(\delta_1) \geq 0) \wedge \cdots \wedge (\mathbb{P}(\delta_{2^n}) \geq 0) \wedge \\
&(\mathbb{P}(\delta_1) + \cdots + \mathbb{P}(\delta_{2^n}) = 1) \wedge \\
&(a_{1,1}\mathbb{P}(\delta_1) + \cdots + a_{1,2^n}\mathbb{P}(\delta_{2^n}) \leq c_1) \wedge \\
&\ldots \wedge \\
&(a_{m,1}\mathbb{P}(\delta_1) + \cdots + a_{m,2^n}\mathbb{P}(\delta_{2^n}) \leq c_m) \wedge \\
&(a'_{1,1}\mathbb{P}(\delta_1) + \cdots + a'_{1,2^n}\mathbb{P}(\delta_{2^n}) > c'_1) \wedge \\
&\ldots \wedge \\
&(a'_{m',1}\mathbb{P}(\delta_1) + \cdots + a'_{m',2^n}\mathbb{P}(\delta_{2^n}) > c'_{m'}) \qquad (2)
\end{aligned}$$

for some integer coefficients $c_1, \ldots, c_m, c_1, \ldots, c'_{m'}, a_{1,1}, \ldots, a_{m,2^n}, a'_{1,1}, \ldots, a'_{m',2^n}$.

Proof. Let $\psi \in \mathcal{L}_{\text{non-prob}}$ be any of the formulas appearing inside of a probability $\mathbb{P}$ in φ. Note that $\mathbb{P}(\psi) = \mathbb{P}(\psi \wedge l_1) + \mathbb{P}(\psi \wedge \neg l_1)$ by Add. Moving on to l_2, we have, provably, $\mathbb{P}(\psi \wedge l_1) = \mathbb{P}(\psi \wedge l_1 \wedge l_2) + \mathbb{P}(\psi \wedge l_1 \wedge \neg l_2)$, and we may rewrite $\mathbb{P}(\psi \wedge \neg l_1)$ similarly. Applying this process successively, we have $\mathbb{P}(\psi) = \mathbb{P}(\psi \wedge \delta_1) + \cdots + \mathbb{P}(\psi \wedge \delta_{2^n})$. For any term in the right-hand side of this inequality, if $\psi \Rightarrow \delta_i$, propositional reasoning by Dist allows us to replace the term by $\mathbb{P}(\delta_i)$, and if not, by 0. Thus we always have that $\mathbb{P}(\psi) = b_1\mathbb{P}(\delta_1) + \cdots + b_{2^n}\mathbb{P}(\delta_{2^n})$ for some coefficients b_i. Applying this process to each $\mathbb{P}$-term in φ and using Ineq to rewrite the left-hand sides of the inequalities, and conjoining the (clearly provable) clauses that $\mathbb{P}(\delta_i) \geq 0$ for all $1 \leq i \leq 2^n$, and $\mathbb{P}(\delta_1) + \cdots + \mathbb{P}(\delta_{2^n}) = 1$, we obtain (2). ◻

The conjunction (2) can be seen as a system of simultaneous inequalities over 2^n unknowns, $\mathbb{P}(\delta_1), \ldots, \mathbb{P}(\delta_{2^n})$. Ineq is actually sound and complete for such systems (we refer the reader to Section 4 of [2] for the proof of this fact). So if φ is consistent with AX—which includes Ineq—this system must have a solution. Thus there are values $\mathbb{P}(\delta_i)$ solving (2). We will now construct a probabilistic simulation model having precisely these probabilities of satisfying each δ_i. Note that for any δ_i with $\models_{\text{non-prob}} \bot \leftrightarrow \delta_i$ it is provable that $\mathbb{P}(\delta_i) = 0$, and we may conjoin this to (2). Note also that $\delta_i \wedge \delta_j$ is unsatisfiable for any $i \neq j$. Given these two observations, the following Lemma implies the result.

Lemma 5. *For any collection of satisfiable $\mathcal{L}_{\text{non-prob}}$-formulas $\varphi_1, \ldots, \varphi_n$ no two of which are jointly satisfiable, and any rational probabilities $p_1, \ldots, p_n \geq 0$ such that $p_1 + \cdots + p_n = 1$, there is a probabilistic simulation model T such that $\mathbb{P}_{\mathsf{T}}(\varphi_i) = p_i$ for all i, $1 \leq i \leq n$.*

Proof. Since the φ_i are satisfiable, there are non-probabilistic simulation models $\mathsf{T}_{\text{non-prob},1}, \ldots \mathsf{T}_{\text{non-prob},n}$ such that for all $i = 1, \ldots, n$, we have $\mathsf{T}_{\text{non-prob},i} \models_{\text{non-prob}} \varphi_i$. Further, we may suppose the machines so constructed use only a bounded number of memory tape squares.[8] Thus let the maximum index of a tape square used by any of the $\mathsf{T}_{\text{non-prob},i}$ be N. We now describe T informally. Suppose without loss of generality that for all i, $p_i = a_i/b$ for some common denominator b. Let T draw a random number r from 1 up to b uniformly, and ensure that T does any auxiliary computations it might need only on squares with indices at least $N + 1$. Check whether $r \leq a_1$, and if so, let T branch into the code of $\mathsf{T}_{\text{non-prob},1}$. If not, check if $a_1 + 1 \leq r \leq a_1 + a_2$ and if so, branch into $\mathsf{T}_{\text{non-prob},2}$. Repeat the process for $p_3, \ldots, p_n$. It's clear that the probability of branching into each $\mathsf{T}_{\text{non-prob},i}$ block is exactly p_i, and the same is true under any relevant (i.e., involving only memory tape variables that appear in one of the φ_i) intervention on T: we may suppose any auxiliary computations T might require use only memory tape squares with indices past N. After branching into the ith block, the behavior of T is exactly the same as that of $\mathsf{T}_{\text{non-prob},i}$, meaning that any random bit tape fixings that end up causing a branch into this block will belong to $S(\varphi_i)$. Another random bit tape fixing that causes a branch into another block, say the jth, cannot belong to $S(\varphi_i)$ since φ_i, φ_j are jointly unsatisfiable. Thus, $\mathbb{P}_{\mathsf{T}}(\phi_i) = p_i$ for all i. □

Finally, we must see that this model lies in $\mathcal{M}^{\downarrow}$ if the original formula is consistent with $\mathsf{AX}^{\downarrow}$. [6] has shown that $\models^{\downarrow}_{\text{non-prob}} [\alpha]\beta \rightarrow \langle\alpha\rangle\beta$. Then in the proof of Lemma 5, we may suppose that each $\mathsf{T}_{\text{non-prob},i}$ block contains only always-halting code,[9] and hence that T does not contain any loops either: thus it almost-surely halts. □

[8] Why? Since φ_i are satisfiable, they are consistent with the axiomatization for non-probabilistic simulation models given by [6], and hence are satisfied by the canonical models given in [6]. These models use only boundedly many tape squares.

[9] Since the canonical programs of [6] for $\mathcal{M}^{\downarrow}_{\text{non-prob}}$ contain only such code.

4 Computational Complexity

Call the problem of deciding if a formula $\varphi \in \mathcal{L}$ is satisfiable Prob-Sim-Sat(φ). Theorem 2 shows that solving this problem is no more complex than is propositional satisfiability.

Theorem 2. Prob-Sim-Sat(φ) *is* NP*-complete in* $|\varphi|$ *(where this length is computed standardly).*

Proof. It's NP-hard since, given any propositional π, the formula $\mathbb{P}(\langle\rangle\pi) > 0$ is satisfiable iff π is satisfiable (consider a machine that does nothing but write a satisfying memory tape assignment out). In order to show that the satisfiability problem is in NP, we give the following nondeterministic satisfiability algorithm: guess a program from a class of programs (that we will define shortly) that includes the program constructed in Lemma 5 —call this canonical program T_φ—and check (in polynomial time) if it satisfies φ. This algorithm decides satisfiability since, by soundness, a satisfiable formula must be consistent, and hence has a canonical model of the form constructed in Lemma 5. For the remainder of the proof, by the "length of a number," we just mean the length of its computer (binary) representation. The "length of a rational" is the sum of the lengths of its numerator and its denominator.

What is the class of probabilistic simulation models that we may limit our guesses to? For some fixed constants $C, D \in \mathbb{N}$, we will define a class $\mathcal{M}_{\varphi,C,D}$. We will then show that there exist C, D such that the canonical program of Lemma 5 belongs to $\mathcal{M}_{\varphi,C,D}$ for all consistent φ. Let $\mathcal{M}_{\varphi,C,D}$ be the fragment of probabilistic simulation models whose code consists of the following:

1. Code to draw a random number uniformly between 1 and some N, such that N has length at most $D|\varphi|^3$.
2. At most $n = C|\varphi|$ *branches*, that is, copies of: an if-statement with condition $\ell \leq r \leq u$, whose body is a canonical program P_{ψ_i} for some $\psi_i \in \mathcal{L}_{\text{non-prob}}$, of the same form as the non-probabilistic canonical models (i.e., in the class defined in the proof of Theorem 2 from [6]).

Letting ℓ_i, u_i be the bounds for the ith copy in (2), we also require that $\ell_1 = 1$, and that $\ell_{i+1} = u_i + 1$ for all i, and that $u_n = N$. The following fact from linear algebra (we refer the reader to [2] for the proof) helps us to show that for all consistent φ, the canonical program T_φ belongs to $\mathcal{M}_{\varphi,C,D}$ for some C, D.

Lemma 6. *A system of m linear inequalities with integer coefficients of length at most ℓ that has a nonnegative solution has a nonnegative solution with at most m variables nonzero, and where the variables have length at most $\mathcal{O}(m\ell + m \log m)$.* □

Apply this lemma to (2). Each inequality in (2) originally came from φ, so there are $\mathcal{O}(|\varphi|)$ of them. Further, recall that each integer coefficient in (2) came from summing up a subset of 2^n coefficients originally from φ, with n is the number of atoms appearing anywhere inside $\mathbb{P}$ expressions in φ. As this n is thus $\mathcal{O}(|\varphi|)$—and hence 2^n is $\mathcal{O}(|\varphi|)$ *in length*—and each original coefficient is also $\mathcal{O}(|\varphi|)$

in length, each coefficient is $\mathcal{O}(|\varphi|)$ in length as well (lengths of products add). Thus Lemma 6 shows that without loss of generality, we may suppose that the solutions for the $\mathbb{P}(\delta_i)$ of (2) have $\mathcal{O}(|\varphi|^2)$ length. The common denominator of these $\mathcal{O}(|\varphi|)$ rationals hence has $\mathcal{O}(|\varphi|^3)$ length. The construction of Lemma 5 has one branch for each of them, and hence $\mathcal{O}(|\varphi|)$ branches. This shows the existence of D for part (1) of the definition of $\mathcal{M}_{\varphi,C,D}$ and the existence of a C for part (2). We will abbreviate $\mathcal{M}_\varphi = \mathcal{M}_{\varphi,C,D}$ for some choice of C, D thus guaranteed.

It remains to show that given any program $\mathsf{T} \in \mathcal{M}_\varphi$, we can check if $\mathsf{T} \models \varphi$ in polynomial time. It suffices to show that checking if $\mathsf{T} \models \psi$ for $\psi \in \mathcal{L}_{\text{ineq}}$ is polynomial time: if we know whether $\mathsf{T} \models \psi$ for every ψ that φ is built out of, we can decide in linear time if $\mathsf{T} \models \varphi$. Thus suppose ψ has the form $a_1\mathbb{P}(\varphi_1) + \cdots + a_n\mathbb{P}(\varphi_n) \leq c$. [6] shows that one may check if the P_{ψ_i} in part (2) of the definition of $\mathcal{M}_\varphi$ satisfy any formula of the basic language $\mathcal{L}_{\text{non-prob}}$ in polynomial time. Then we can easily compute $\mathbb{P}(\varphi_i)$ as simply the sum of the probabilities of each branch that satisfies φ_i. Doing the arithmetic to check if ψ is satisfied is then certainly polynomial time, so we have our result. □

5 Conclusion and Future Work

We have defined and obtained foundational results concerning a very natural extension of counterfactual intervention on simulation models to the probabilistic case.

One critical operation in probability is *conditioning*, or updating probabilities given that some event is known to have occurred (in the subjective interpretation, updating a belief for known information). One may already define conditional probabilities in the usual way in the current framework, and our framework (without interventions) covers the *conditional simulation* approach to certain aspects of common-sense reasoning of [3]. In this approach, one limits oneself to the runs satisfying a certain query; the framework considered here would be equivalent for any queries expressible as formulas of $\mathcal{L}_{\text{non-prob}}$. [2] also give a logic for reasoning about conditional probabilities. Future work would involve extending this system to probabilistic simulation models and studying the complexity of reasoning in that setting.

As [8, 6] note, the simulation model approach invalidates many important logical principles that are valid in other approaches [5, 11, 9], such as *cautious monotonicity*: $[A](B \wedge C) \rightarrow [A \wedge B]C$. However the approach is otherwise quite general, and an important future direction would be to identify and characterize subclasses of of simulation models that validate this and other similar logical principles. We have begun investigating this extension. An interesting consequence it has is on the comparison of conditional probability with the probabilities of subjunctive conditionals: while these two probabilities are not in general equal in the classes $\mathcal{M}$ or $\mathcal{M}^{\downarrow}$, they are equal in certain restricted classes.

A final direction we want to mention concerns "open-world" reasoning including first-order reasoning about models with some domain, where counterfactual

antecedents might alter how many individuals are being considered or which individuals fall under a property or bear certain relations to each other. Recursion and the tools of logic programming [4, 10] make this very natural for the simulation model approach, and we would like to understand the first- and higher-order conditional logics that result in this approach, in both the non-probabilistic and probabilistic cases. We have also begun exploring this direction.

References

1. Chater, N., Oaksford, M.: Programs as causal models: Speculations on mental programs and mental representation. Cognitive Science **37**(6), 1171–1191 (2013)
2. Fagin, R., Halpern, J.Y., Megiddo, N.: A logic for reasoning about probabilities. Information and Computation **87**, 78–128 (1990)
3. Freer, C.E., Roy, D.M., Tenenbaum, J.B.: Towards common-sense reasoning via conditional simulation: legacies of turing in artificial intelligence. In: Downey, R. (ed.) Turing's Legacy: Developments from Turing's Ideas in Logic, Lecture Notes in Logic, vol. 42, pp. 195–252. Cambridge University Press (2014)
4. Goodman, N.D., Tenenbaum, J.B., Gerstenberg, T.: Concepts in a probabilistic language of thought. In: Margolis, E., Laurence, S. (eds.) The Conceptual Mind: New Directions in the Study of Concepts. MIT Press (2015)
5. Halpern, J.Y.: Axiomatizing causal reasoning. Journal of AI Research **12**, 317–337 (2000)
6. Ibeling, D., Icard, T.: On the conditional logic of simulation models. Proc. 27th IJCAI (2018)
7. Icard, T.: Beyond almost-sure termination. Proc. 39th CogSci (2017)
8. Icard, T.F.: From programs to causal models. In: Cremers, A., van Gessel, T., Roelofsen, F. (eds.) Proceedings of the 21st Amsterdam Colloquium. pp. 35–44 (2017)
9. Lewis, D.: Counterfactuals. Harvard University Press (1973)
10. Milch, B., Marthi, B., Russell, S., Sontag, D., Ong, D.L., Kolobov, A.: BLOG: Probabilistic models with unknown objects. In: Proc. 19th IJCAI. pp. 1352–1359 (2005)
11. Pearl, J.: Causality. CUP (2009)