Decidability and Expressiveness for First-Order Logics of Probability

Martín Abadi and Joseph Y. Halpern

October 2, 1996

A preliminary version of this report appeared in the proceedings of the 30th Annual Symposium on Foundations of Computer Science, held in Research Triangle Park, North Carolina, USA, in October 1989. This version is almost identical to one that appears in *Information and Computation* **112**:1, 1994, pp. 1–36.

Joseph Y. Halpern is at the IBM Almaden Research Center, 650 Harry Road, San Jose, California 95120, USA.

Authors' Abstract

We consider decidability and expressiveness issues for two first-order logics of probability. In one, the probability is on possible worlds, while in the other, it is on the domain. It turns out that in both cases it takes very little to make reasoning about probability highly undecidable. We show that when the probability is on the domain, if the language contains only unary predicates then the validity problem is decidable. However, if the language contains even one binary predicate, the validity problem is Π_1^2 complete, as hard as elementary analysis with free predicate and function symbols. With equality in the language, even with no other symbol, the validity problem is at least as hard as that for elementary analysis, Π_∞^1 hard. Thus, the logic cannot be axiomatized in either case. When we put the probability on the set of possible worlds, the validity problem is Π_1^2 complete with as little as one unary predicate in the language, even without equality. With equality, we get Π_∞^1 hardness with only a constant symbol. We then turn our attention to an analysis of what causes this overwhelming complexity. For example, we show that if we require rational probabilities then we drop from Π_1^2 to Π_1^1. In many contexts it suffices to restrict attention to domains of bounded size; fortunately, the logics are decidable in this case. Finally, we show that, although the two logics capture quite different intuitions about probability, there is a precise sense in which they are equi-expressive.

Contents

1 Introduction

Reasoning about probability is crucial in many contexts, from analyzing probabilistic programs to reasoning about uncertainty in expert systems. Especially in the context of expert systems, there is a great deal of interest in finding a language appropriate for carrying out such reasoning, and then automating it. Recently there has been much research in the area of probabilistic reasoning about propositional statements: it is possible to provide straightforward syntax and semantics for rich propositional languages for reasoning about probability, with relatively tractable decision procedures and complete axiomatizations [FHM90, GKP88, Nil86].

Propositional logic is often not expressive enough to capture many important situations; we would like to have the machinery of first-order logic, with functions, predicates, and quantification. It thus becomes of interest to extend the results we have for the propositional case to the first-order case. Of course, a language for reasoning about probability ought to have easily comprehensible syntax and semantics. Ideally, the validity problem would be not much worse than that for first-order logic, and we would have a complete axiomatization that could provide some guidance for automating the reasoning process.

Unfortunately, even providing semantics for a first-order logic of probabilities is not completely straightforward. As pointed out by Bacchus [Bac88, Bac90], the possible-worlds semantics used in [Nil86] and (for the propositional case) in [FHM90] is not expressive enough. To understand the problem, consider the two statements "The probability that Tweety flies is greater than .9" and "The probability that a randomly chosen bird flies is greater than .9".

We can capture the first sentence using the possible worlds approach in a straightforward way. We consider a number of possible worlds. Tweety flies in some of them, but not in others. (Thus, the predicate *Fly* has different extensions in different possible worlds.) We put a probability on the space of possible worlds, and then say that the probability that Tweety flies is greater than .9 exactly if the set of worlds where *Fly(Tweety)* holds has probability greater than .9.

This approach will not serve to capture the statistical information inherent in the second statement. In particular, it does not correspond to saying that the set of worlds where the statement $\forall x(Bird(x) \Rightarrow Fly(x))$ holds has probability greater than .9. We might believe that 90% of all birds fly without believing that there is any possible world where all birds fly. Bacchus shows that other attempts to capture the second statement using a possible-worlds approach suffer from similar flaws.

Intuitively, the first statement suggests the existence of a number of possible worlds, with a probability on the set of worlds, while the second seems to assume only one possible world (the "real" world), with a probability on the domain (so that if we pick a bird at random from the domain, it will fly with probability greater than .9). In [Hal90], this situation is analyzed in detail, and syntax and semantics are provided for two logics, one of which assumes a probability on the domain, while the other one assumes a probability on the possible worlds.

In this paper, we consider the complexity of the validity problem for these logics, and investigate how formulas of the first can be used to capture ideas of the second, and vice versa.

For classical predicate calculus, it is known that the validity problem is decidable if the language has only unary predicates, but the validity problem is undecidable with even one binary predicate [DG79, Lew79]. However, no matter what predicates and functions we allow in the language, the validity problem is recursively enumerable; moreover, there is a well-known complete axiomatization for first-order logic [End72]. We show that the validity problem with just unary predicates is decidable for the first logic of probability. (A complete axiomatization is provided in [Hal90].) However, once we add even one binary predicate, the logic becomes wildly undecidable, in fact complete for Π^2_1. This is the complexity of the validity problem for elementary analysis with free predicate and function symbols; it is even harder than the complexity of elementary analysis (or equivalently, second-order arithmetic with set variables) which is Π^1_∞ (see [Rog67, Hin78] for details). Of course, it follows that there cannot be a complete axiomatization for the language.

The situation is even worse if we have equality. For first-order logic with equality, the validity problem with only unary predicates in the language is still decidable [DG79]. However, with equality and probability in the picture, we can get Π^1_∞ hardness without any predicates and functions in the language at all!

In the case of the second logic, where the probability is over the possible worlds, we get Π^2_1 completeness as soon as we have even one unary predicate. If equality is included, we get Π^1_∞ hardness as long as there is at least one constant symbol.

Roughly speaking, our undecidability results say that as long as Φ is "sufficiently rich", the validity problem for first-order reasoning about probability is wildly undecidable. Exactly which richness condition is required for Φ and how undecidable the logic is depends on whether we are considering probability on the domain or on possible worlds, and whether or not equality is in the language. The situation is summarized in the table below, where only lower bounds on complexity are stated; $\mathcal{L}_1(\Phi)$ and $\mathcal{L}_2(\Phi)$ denote the first and the second logics, with the set of function and predicate symbols Φ, and $\mathcal{L}_1^=(\Phi)$ and $\mathcal{L}_2^=(\Phi)$ denote the same logics with equality.

language	richness condition on Φ	complexity
$\mathcal{L}_1(\Phi)$	at least one binary predicate	Π_1^2
$\mathcal{L}_1^=(\Phi)$	—	Π_∞^1
$\mathcal{L}_2(\Phi)$	at least one unary predicate	Π_1^2
$\mathcal{L}_2^=(\Phi)$	at least one constant symbol	Π_∞^1

Table 1: Undecidability for first-order logics of probability

These results stand in contrast to those of Bacchus [Bac90], who provides a complete axiomatization for a logic with essentially the same syntax as our first logic, with probability on the domain. What allows Bacchus to obtain his result (which in particular shows that the validity problem for his language is semi-decidable) is that he allows *nonstandard* probability functions, which are only required to be finitely additive (rather than countably additive) and he allows probabilities to take values in arbitrary ordered fields.

Bacchus's result motivates us to consider exactly what it is that makes reasoning about probability so highly intractable. For example, it turns out that all of our undecidability results go through without change if we allow probabilities to be finitely additive. We can get Π_1^1 hardness results without having multiplication in the language or quantification over the reals. In fact, we already get Π_1^1 hardness if we restrict probabilities to taking rational values. There is some good news in this bleak picture: if we restrict attention to domains of bounded size (which arise frequently in AI applications), then both logics are decidable.

Finally, we consider the issue of the expressive power of these two logics. Although the logics capture very different intuitions about probability, we show that in a precise sense they are equally expressive. There exists a translation taking a structure M for the first logic to a structure M' for the second, and taking a formula φ of the first logic to a formula φ' of the second such that $M \models \varphi$ iff $M' \models \varphi'$. Similar results hold in translating from the second logic to the first.

Although work relating first-order logic and probability goes back to Carnap [Car50], there has been relatively little work on providing formal first-order logics for reasoning about probability. Besides the work of Bacchus mentioned above, Gaifman [Gai60, Gai64] considered the problem of associating probabilities with classical first-order statements (which, as pointed out in [Bac88], essentially corresponds to putting probabilities on possible worlds); Hoover [Hoo78] considered questions related to such a logic. Łoś and Fenstad studied this problem as well, but allowed values for free variables to be chosen according to a probability on the domain [Łoś63, Fen67]. Gaifman and Snir [GS82] used a logic where probabilities are put on sentences to investigate issues related to randomness. Keisler [Kei85] investigated an infinitary logic with a measure on the domain, and obtained completeness and compactness results. Feldman and Harel [FH84, Fel84] considered a probabilistic dynamic logic, which extends first-order dynamic logic by adding probability. There are commonalities between the program-free fragment of Feldman and Harel's logic and our logics, but since their interest is in reasoning about probabilistic programs, their formalism is significantly more complex than ours, and they focus on proving that their logic is complete relative to its program-free fragment.

The rest of this paper is organized as follows. In the next three sections, we review the syntax and semantics of the two logics (using material taken from [Hal90]) and then some facts about the arithmetical and the analytical hierarchies. In Section 5, we discuss our decidability and undecidability results. We consider expressiveness issues in Section 6.

2 Probabilities on the domain

We assume that we have a first-order language for reasoning about some domain. We take this language to consist of a collection Φ of predicate symbols and function symbols of various arities. (As usual, we can identify constant symbols with function symbols of arity 0.) Given a formula φ in the logic, we also allow formulas such as $w_x(\varphi) \geq 1/2$, which can be interpreted as "the probability that a random x in the domain satisfies φ is greater than or equal to 1/2". We actually extend this to allow arbitrary sequences of distinct variables in the subscript. To understand the intuition behind this, suppose the formula $Son(x, y)$ says that x is the son of y. Now consider the three terms $w_x(Son(x, y))$, $w_y(Son(x, y))$, and $w_{\langle x,y\rangle}(Son(x, y))$. The first describes the probability that a random x is the son of y; the second describes the probability that x is the son of a random y; the third describes the probability that a random pair (x, y) will have the property that x is the son of y.

We formalize these ideas by using a two-sorted language. The first sort consists of the function symbols and predicate symbols in Φ, together with a countable family of *object variables* $x^o, y^o, \ldots$. The second sort consists of the constant symbols 0 and 1, the binary function symbols $+$ and $\times$, the binary relation symbols $>$ and $=$, and a countable family of *field variables* $x^f, y^f, \ldots$. (We drop the superscripts on the variables when it is clear from context of what sort they are.) We form object terms, which range over the domain of the first-order language, by starting with object variables and closing off under function application, so that if f is an n-ary function symbol in Φ and $t_1, \ldots, t_n$ are object terms, then $f(t_1, \ldots, t_n)$ is an object term. We then define formulas and field terms simultaneously; field terms range over the reals (or, later subfields of the reals). Informally, field terms are formed by starting with 0, 1, and probability terms of the form $w_{\vec{x}}(\varphi)$, where φ is a formula, and closing off under $+$ and $\times$, so that $t_1 + t_2$ and $t_1 \times t_2$ are field terms if t_1 and t_2 are. We form formulas in the standard way. We start with *atomic formulas*: if P is an n-ary predicate symbol in Φ, and $t_1, \ldots, t_n$ are object terms, then $P(t_1, \ldots, t_n)$ is an atomic formula, while if t_1 and t_2 are field terms then $t_1 = t_2$ and $t_1 > t_2$ are atomic formulas. We also consider the situation where there is an equality symbol for object terms; in this case, if t_1 and t_2 are object terms, then $t_1 = t_2$ is also an atomic formula. We then close off under conjunction, negation, and universal quantification, so that if φ_1 and φ_2 are formulas and x is a (field or object) variable, then $\varphi_1 \wedge \varphi_2$, $\neg\varphi_1$, and $\forall x \varphi_1$ are also formulas. A formal definition of field terms and formulas can be given by induction on the depth of nesting of expressions of the form $w_{\vec{x}}(\varphi)$ that appear in field terms; we omit details here. We call the resulting language $\mathcal{L}_1(\Phi)$; if it includes equality between object terms, we call it $\mathcal{L}_1^{=}(\Phi)$.

We define $\vee$, $\Rightarrow$, and $\exists$ in terms of $\wedge$, $\neg$, and $\forall$ as usual. In addition, if t_1 and t_2 are two field terms, we use other standard abbreviations, such as $t_1 \geq t_2$ for $(t_1 > t_2) \vee (t_1 = t_2)$ and $t_1 \geq 1/2$ for $(1 + 1) \times t_1 \geq 1$.

The only differences between our syntax and that of Bacchus is that we write $w_{\vec{x}}(\varphi)$ rather than $[\varphi]_{\vec{x}}$, and, for simplicity, we do not consider what Bacchus calls *measuring functions* (functions which map object terms into field terms), and the only field functions we allow are $+$ and $\times$. The language is still quite rich, allowing us to express conditional probabilities, notions of independence, and statistical notions; we refer the reader to [Bac90] for examples.

We define a *type 1 probability structure over* Φ to be a tuple (D, π, μ), where D is a domain, π assigns to the predicate and function symbols in Φ predicates and functions of the right arity over D, and μ is a discrete probability function on D. That is, we take μ to be a mapping from D to the real interval $[0,1]$ such that $\sum_{d\in D}\mu(d) = 1$. For any $A \subseteq D$, we define $\mu(A) = \sum_{d\in A}\mu(d)$. (The restriction to discrete probability functions is made here mainly for ease of exposition, and does not affect any of our lower-bound results; we do, however, need discreteness for our upper-bound results.) We can then define a discrete probability function μ^n on the product domain D^n consisting of all n-tuples of elements of D by taking $\mu^n(d_1, \ldots, d_n) = \mu(d_1) \times \ldots \times \mu(d_n)$. Define a *valuation* to be a function mapping each object variable into an element of D and each field variable into an element of $I\!R$ (the reals). Given a type 1 probability structure M and valuation v, we proceed by induction to associate with every object (resp. field) term t an element $[t]_{(M,v)}$ of D (resp. $I\!R$), and with every formula φ a truth value, writing $(M, v) \models \varphi$ if the value true is associated with φ by (M, v). The definition follows the lines of the corresponding one for the classical predicate calculus, so we just give a few clauses of the definition here:

- $(M, v) \models t_1 = t_2$ iff $[t_1]_{(M,v)} = [t_2]_{(M,v)}$;
- $(M, v) \models \forall x^o \varphi$ iff $(M, v[x^o/d]) \models \varphi$ for all $d \in D$, where $v[x^o/d]$ is the valuation which is identical to v except that it maps x^o to d;
- $[w_{\langle x_1,\ldots,x_n\rangle}(\varphi)]_{(M,v)} = \mu^n(\{(d_1, \ldots, d_n) : (M, v[x_1/d_1, \ldots, x_n/d_n]) \models \varphi\})$.

We write $M \models \varphi$ if $(M, v) \models \varphi$ for all valuations v, and say that φ *is valid with respect to type 1 structures*, if $M \models \varphi$ for all type 1 probability structures M.

3 Probabilities on possible worlds

The syntax for a logic for reasoning about possible worlds is essentially the same as the syntax used in the previous section. Starting with a set Φ of predicate and function symbols, we form more complicated formulas and terms as before, except that instead of allowing probability terms of the form $w_{\vec{x}}(\varphi)$, where $\vec{x}$ is some vector of distinct object variables, we only allow probability terms of the form $w(\varphi)$, interpreted as "the probability of φ". Since we are no longer going to put a probability distribution on the domain, it does not make sense to talk about the probability that a random choice for $\vec{x}$ will satisfy φ. It does make sense to talk about the probability of φ, though: this will be the probability of the set of possible worlds where φ is true. We call the resulting language $\mathcal{L}_2(\Phi)$; if it includes equality between object terms, we call it $\mathcal{L}_2^=(\Phi)$.

More formally, a *type 2 probability structure over* Φ is a tuple (D, S, π, μ), where D is a domain, S is a set of *states* or *possible worlds*, $\pi(s)$ assigns to the predicate and function symbols in Φ predicates and functions of the right arity over D for each state $s \in S$, and μ is a discrete probability function on S. Thus, each state $s \in S$ can be viewed as a first-order structure. All these structures have a common domain, namely D, but they may differ in the interpretations they assign to the function and predicate symbols in Φ. Roughly speaking, in order to evaluate the field term $w(\varphi)$, we fix an assignment v of values in D to the free variables of φ, and then compute the probability of the set of states where φ is true under assignment v. Note the key difference between type 1 and type 2 probability structures: in type 1 probability structures, the probability is taken over the domain D, while in type 2 probability structures, the probability is taken over the set S of states.

Given a type 2 probability structure M, a state s, and valuation v, we can associate with every object (resp. field) term t an element $[t]_{(M,s,v)}$ of D (resp. $I\!R$), and with every formula φ a truth value, writing $(M, s, v) \models \varphi$ if the value true is associated with φ by (M, s, v). We now need the state on the left-hand side of $\models$ to provide interpretations for the predicate and function symbols. (Recall that they might have different interpretations in each state.) Again, we just give a few clauses of the definition here, to indicate the similarities and differences between type 1 and type 2 probability structures:

- $(M, s, v) \models P(x)$ iff $v(x) \in \pi(s)(P)$;
- $(M, s, v) \models t_1 = t_2$ iff $[t_1]_{(M,s,v)} = [t_2]_{(M,s,v)}$;
- $(M, s, v) \models \forall x^o \varphi$ iff $(M, s, v[x^o/d]) \models \varphi$ for all $d \in D$;
- $[w(\varphi)]_{(M,s,v)} = \mu(\{s' \in S : (M, s', v) \models \varphi\})$.

We write $(M, s) \models \varphi$ if $(M, s, v) \models \varphi$ for all valuations v. Similarly, we write $M \models \varphi$ if $(M, s) \models \varphi$ for all states s in M. We say that φ is *valid with* *respect to type 2 structures* if $M \models \varphi$ for all type 2 probability structures M.

4 Some complexity classes

In our proofs, we show that, in most cases, both logics of probability are expressive enough to allow systems of arithmetic and analysis to be encoded. Hence, the validity problems for our logics are highly undecidable; in particular, the logics are not axiomatizable. In this section, we give definitions of classes such as Π^1_1, Π^1_∞, and Π^2_1 that come into the study of variants of our logics, and we describe properties of these classes that will be important in our proofs. Our treatment here is quite sketchy; the interested reader should consult the books by Rogers and Hinman [Rog67, Hin78] for more details.

A Π^0_∞ *formula* is a first-order formula with nonlogical symbols 0, 1, +, and $\times$ (and with equality), and variables $x_1, x_2, \ldots$ which, intuitively, range over natural numbers. A Π^0_∞ formula is one in what has commonly been called the language of arithmetic. Gödel's famous incompleteness result shows that the set of Π^0_∞ *sentences* (that is, Π^0_∞ formulas with no free variables) that are true when interpreted over the natural numbers cannot be characterized by a recursive set of axioms. To prove this, it would suffice to show that this set of formulas is not recursively enumerable (r.e.). In fact, it is much harder than r.e.; it is what we shall call Π^0_∞ *complete.*

A Π^0_∞ *formula with second-order parameters* is a Π^0_∞ formula, except that there may appear *set variables* $X_1, X_2, \ldots$ in expressions of the form $t \in X_i$ (where t is a term with no set variables). Intuitively, these set variables range over sets of numbers. Similarly, a Π^0_∞ *formula with second-order and third-order parameters* is a Π^0_∞ formula, except that there may appear set variables and *set of sets variables* $\mathcal{X}_1, \mathcal{X}_2, \ldots$ in expressions of the form $X_i \in \mathcal{X}_j$. Intuitively, the set of sets variables range over sets of sets of numbers.

A Π^1_1 *formula* is a Π^0_∞ formula with second-order parameters preceded by universal second-order quantifiers that bind second-order parameters (the set variables). Thus, a Π^1_1 formula has the form $\forall X_1 \forall X_2 \ldots \forall X_n \varphi$, where φ is a Π^0_∞ formula with second-order parameters and $X_1, \ldots, X_n$ are second-order variables. A Π^1_∞ *formula* is a Π^0_∞ formula with second-order parameters preceded by arbitrary second-order quantifiers. The Π^1_∞ formulas give rise to a language commonly known as *second-order arithmetic with set vari-*

ables. A Π^2_1 *formula* is a Π^1_∞ formula with third-order parameters, preceded by universal third-order quantifiers.

A dual set of Σ^j_i formulas corresponds to each set Π^j_i that we have defined. For example, a Σ^2_1 *formula* is a Π^0_∞ formula with second- and third-order parameters, preceded by arbitrary second-order quantifiers and existential third-order quantifiers. We can also define the classes Σ^1_n and Π^1_n for $n = 0, 1, 2, \ldots$ by taking $\Sigma^1_0 = \Pi^1_0 = \Pi^0_\infty$ and defining the Σ^1_{n+1} formulas to consist of Π^1_n formulas preceded by existential second-order quantifiers and the Π^1_{n+1} formulas to consist of Σ^1_n formulas preceded by universal quantifiers. Generally, in an expression such as Π^i_n, the superscript tells us the order of the parameters (if the superscript is i, we have parameters of order $i+1$) and the subscript tells us the type of quantification that we are allowed over these parameters.

For each of these classes of formulas, there is a corresponding set of sets of natural numbers and a corresponding set of problems that can be defined with the formulas of that class. For example, the set S is a Π^1_1 set if there exists a Π^1_1 formula $\varphi_S(x)$ with one free variable x such that $n \in S$ if and only if $\varphi_S(n)$ holds; the corresponding Π^1_1 problem consists of determining whether $n \in S$.

We identify each class of formulas with a corresponding class of codes for them, in some standard encoding; the truth problem for a given class is the problem of deciding whether a sentence in that class is true, or more precisely whether the code for the sentence is a member of the set of codes for true sentences. It is easy to see that the truth problem for Π^1_1 sentences is Π^1_1 hard, since if S is a Π^1_1 set and n is a number we can decide whether $n \in S$ by deciding whether $\varphi_S(n)$ is true. Similarly, the truth problem for Π^1_∞ sentences is Π^1_∞ hard, and the truth problem for Π^2_1 sentences is Π^2_1 hard. With a bit more work, one can obtain normal-form theorems, and then show that the truth problem for Π^1_1 sentences with just one second-order quantifier is Π^1_1 hard. Similarly, the truth problem for Π^2_1 sentences with just one third-order quantifier is Π^2_1 hard.

Conversely, the sets of true Π^1_1 and Π^2_1 sentences are Π^1_1 and Π^2_1 sets, respectively. It follows that the truth problems for Π^1_1 and Π^2_1 sentences are complete for the corresponding classes. On the other hand, the set of true Π^1_∞ sentences is not a Π^1_∞ set, since if it were it would have to be in Π^1_n for some n, and it is not hard to show that this is not the case. (More generally, it can be shown that no set in Π^1_∞ can be Π^1_∞ hard.) Nevertheless, we say that this set is Π^1_∞ *complete*, and we also call Π^1_∞ complete all the sets recursively equivalent to it. Intuitively, this is justified because the set is Π^1_∞ hard, and can be written as the recursive union of Π^1_n sets. Thus, although the set is not in Π^1_∞, in some sense it is just beyond. Similarly, we can obtain the notion of Π^0_∞ completeness.

There are actually many equivalent ways of defining these complexity classes. In particular, there are many equivalent ways to present higher-order quantification. Above, we have used variables that range over sets of numbers and sets of sets of numbers. Through standard coding techniques it is simple to show that we could also have quantified over predicates and functions over the naturals, instead of over sets of natural numbers, and then over functionals (functions from functions to functions), instead of over sets of sets. Notice that it does not matter whether the functions in question return naturals, reals, or sets of naturals. For example, a function f that takes a natural n and returns a real $f(n)$ is equivalent to a function that takes two naturals n and k and returns the k^{th} bit of the binary expansion of $f(n)$.

Another way to arrive at the same definitions is by replacing sets of natural numbers with real numbers (and, correspondingly, sets of sets of natural numbers with sets of real numbers). In this approach, naturals and reals are kept separate, in the sense that variables that range over all reals are distinguished from those that range over natural numbers. On the other hand, it is permitted to add and to multiply reals, and it is also permitted to say that a real equals a natural. In this fashion, we obtain systems of analysis. In particular, the system known as *elementary analysis* is obtained as the analogue of second-order arithmetic with set variables, and as a matter of fact elementary analysis and second-order arithmetic with set variables have exactly the same power.[1]

Yet another variant consists in not including multiplication of natural numbers in the language. By the time we have universal quantification over sets of natural numbers (as we do once we are at the Π^1_1 level), having multiplication in the language does not add expressive power, since we can define the multiplication relation. (The idea is to first define the set of perfect squares using a universal second-order quantifier and addition (as in [Hal91]), and then define multiplication exploiting the identity $(m+n)^2 =$

[1]In Rogers' description of analysis, the nonnegative reals are the only ones considered. We typically work with all reals, but this is only a superficial difference. Trivially, a system with all reals is at least as expressive as one with the nonnegative reals (since we can say that a real is nonnegative). Conversely, any statement about the reals can be transformed into a statement about nonnegative reals. The idea is to encode a real as a pair of nonnegative reals; we omit details here.

$m^2 + 2mn + n^2$.)

Finally, all of these devices can be mixed, and for example the same Π^1_∞ sets are obtained if we allow quantification over reals and over sets of natural numbers at once.

5 Decidability and undecidability results

In this section we consider the complexity of the validity problem. The structures we are interested in contain the reals with addition and multiplication. The first-order theory of the reals with addition and multiplication is well known to be equivalent to the theory of real closed fields and to be decidable [Tar55]. (In fact, it is decidable in exponential space, by results of [BKR86].) This might give us some hope that our languages might have relatively tractable decision procedures. This hope is realized in one special case, namely, for $\mathcal{L}_1(\Phi)$ with unary predicates (see Theorem 5.1 below). However, as we mentioned in the introduction, the validity problem is highly undecidable in general. Some intuition behind this might stem from the observation that, although the theory of the reals with addition and multiplication is decidable, once we have some additional structure, such as a predicate defining the natural numbers (so that we can effectively quantify over both the reals and the naturals), we then get to Π^1_∞, the level of second-order arithmetic. As our results below show, once we have a binary predicate in the language, for example, we can do enough encoding to get us to this level and beyond, using the probability functions.

We start with $\mathcal{L}_1(\Phi)$ and type 1 structures. It is well known that the validity problem for first-order logic where the language has no function symbols (other than constants) and only unary predicates is decidable [DG79]. Thus, the following result might not seem too surprising.

Theorem 5.1 : *If Φ consists only of unary predicates, then the validity problem for $\mathcal{L}_1(\Phi)$ with respect to type 1 probability structures is decidable.*

Proof: As shown in [Hal90, Theorem 5.7, Claim 2], if Φ consists of monadic predicates only and there is no equality, for any closed formula φ we can effectively find closed formulas φ_1 and φ_2 such that:

- φ is valid iff both φ_1 and φ_2 are valid,
- φ_1 is a pure first-order formula over Φ (and so is formed from the symbols in Φ and object variables, using first-order quantification), and

- φ_2 is a formula in the language of real closed fields (and so is formed from 0, 1, $+$, $\times$, $>$, $=$, and field variables, using first-order quantification over field variables).

The result now follows from the decidability of the theory of real closed fields and the decidability of first-order logic with only unary predicates. ∎

Once we allow even one binary predicate into Φ, first-order logic becomes undecidable [DG79], although it is recursively enumerable (no matter how rich Φ is), and has an elegant complete axiomatization [End72]. Unfortunately, the situation gets much worse once we allow reasoning about probabilities. As we mentioned above, the probability functions allow us to do sufficient encoding to get a high degree of undecidability. In fact, as soon as we allow even one binary predicate into the language, the validity problem becomes Π^2_1 complete.

Theorem 5.2: *If Φ contains at least one predicate of arity greater than or equal to two, then the validity problem for $\mathcal{L}_1(\Phi)$ with respect to type 1 probability structures is Π^2_1 complete.*

Proof: We first prove the lower bound. Suppose we have a binary predicate B in Φ. Consider an arbitrary Σ^2_1 sentence ψ. We show how to construct a formula ψ' in $\mathcal{L}_1(\{B\})$ such that ψ is true iff ψ' is satisfiable. This will prove the lower bound.

Intuitively, we are going to force the domain to be the disjoint sum of the natural numbers with its power set. We first separate the domain into two components, the elements x such that $B(x,x)$, and those such that $\neg B(x,x)$. Intuitively, we can think of elements in the first component (or, more accurately, equivalence classes of elements in the first component) as representing the natural numbers, while those in the second represent sets of natural numbers. We also use elements in the first component to represent sets of sets of natural numbers. In addition, if x is in the first component and y is in the second, then $B(x,y)$ holds exactly if the number represented by (the equivalence class of) x is in the set represented by y, while $B(y,x)$ holds if the set represented by y is in the set of sets represented by x.

Let ψ_1 be a formula that forces B to be an equivalence relation on the first component:

$$\begin{aligned}\psi_1 =_{\rm def}\ & \forall x,y,z((B(x,x)\wedge B(y,y)\wedge B(z,z))\\ & \Rightarrow\ (B(x,y)\Rightarrow B(y,x))\wedge(B(x,y)\wedge B(y,z)\Rightarrow B(x,z))).\end{aligned}$$

The formula ψ_2 says that if an element x in the first component is related to y via B, then so are all the other elements in x's equivalence class:

$$\psi_2 =_{\rm def} \forall x, x', y(B(x,x) \wedge B(x',x') \wedge B(x,x') \wedge B(x,y) \Rightarrow B(x',y)).$$

Now consider the formula

$$\begin{array}{ll}\psi_3 =_{\rm def} & \exists z(B(z,z) \wedge w_x(B(x,z)) = 1/2) \wedge \\ & \forall y(B(y,y) \Rightarrow \exists y'(B(y',y') \wedge 2w_x(B(x,y')) = w_x(B(x,y)))).\end{array}$$

The formula ψ_3 requires that there be equivalence classes with probabilities $1/2$, $1/4$, $1/8$, $\ldots$ in the first component of the domain. Since the sum of the probabilities of these equivalence classes is 1, these can be the only equivalence classes of positive probability in the domain. It follows that all the elements of positive probability in the domain are in the first component.

The following formula forces every equivalence class in the first component to have positive probability:

$$\psi_4 =_{\rm def} \forall y(B(y,y) \Rightarrow w_x(B(x,y)) > 0).$$

Thus, we have a natural bijection between equivalence classes in the first component and the natural numbers. We can identify the natural number n with the equivalence class consisting of elements d in the natural number component such that $w_x(B(x,d)) = 1/2^{n+1}$. (This technique for encoding the natural numbers is due to Feldman [Fel84]; we thank him for pointing it out to us.)

We encode sets of natural numbers in the second component. There is a straightforward way of doing this. For an element d in the first component, let $Nat(d)$ be the natural number identified with the equivalence class of d. Given an element e in the second component, we can associate it with the set consisting of $Nat(d)$ for each element d in the first component such that $B(d,e)$ holds. The next formula ensures that every set of natural numbers can be represented in this way. This, in turn, is done by ensuring that for each real number r between 0 and 1 there is an element y in the second component such that the set of x's in the first component for which $B(x,y)$ holds has probability r.

$$\psi_5 =_{\rm def} \forall r(0 \leq r \leq 1 \Rightarrow \exists y(\neg B(y,y) \wedge w_x(B(x,y)) = r)).$$

A priori, it is not clear that ψ_5 is satisfiable. It may be impossible to find a subset of the first component whose probability is r, for all r between 0 and 1. To see that it is possible, given a real number $r \in [0,1]$, consider a binary representation of r and let A be the union of all equivalence classes representing those natural numbers n such that $r_{n+1} = 1$, where r_i is the i^{th} bit in the binary representation. It follows from the identification above that the probability of A is precisely r. Essentially, the formula ψ_5 forces there to be an element e in the second component such that $B(d,e)$ holds iff $d \in A$. As we mentioned, we can then identify e with the set $\{Nat(d) : d \in A\}$. Under this identification, we can informally think of $B(d,e)$ as saying "$d \in e$".

We would now like to argue that since every set of natural numbers can be associated in an obvious way with a binary representation of a real number, it follows that every set of natural numbers can be represented by an element in the second component. There is a slight technical problem with this argument: a given real number may represent two different sets of natural numbers. For example, the singleton set $\{0\}$ is represented by .1, and its complement is represented by .01111..., but both of these are representations of the real number 1/2. This problem is easily seen to occur only if the binary representation of a set has either finitely many 0's or finitely many 1's (that is, if the set is either finite or cofinite). Thus, we add a formula that "says" that every finite and cofinite set is represented. We proceed as follows. If y is an element of the second component, the formula

$$\begin{aligned}\theta_1(y) =_{\text{def}} \;& \exists x(B(x,x) \wedge B(x,y)) \wedge \\ & \exists r > 0(\forall x(B(x,x) \wedge w_z(B(z,x)) < r \Rightarrow \neg B(x,y)))\end{aligned}$$

expresses that y represents a nonempty and finite set. Note that the way we say that y represents a finite set is to say that it does not contain any representations of natural numbers with probability greater than some fixed $r > 0$. Similarly, the formula

$$\begin{aligned}\theta_2(y) =_{\text{def}} \;& \exists x(B(x,x) \wedge \neg B(x,y)) \wedge \\ & \exists r > 0(\forall x(B(x,x) \wedge w_z(B(z,x)) < r \Rightarrow B(x,y)))\end{aligned}$$

expresses that the complement of the set represented by y is nonempty and finite. If y' is also an element of the second component, the formula

$$\theta_3(y,y') =_{\text{def}} (w_x(B(x,y)) = w_x(B(x,y'))) \wedge \exists x(B(x,x) \wedge B(x,y') \wedge \neg B(x,y))$$

expresses that y and y' represent different sets but correspond to the same real number. We put all these pieces together and obtain the following formula:

$$\psi_6 =_{\text{def}} \forall y(\neg B(y,y) \wedge (\theta_1(y) \vee \theta_2(y)) \Rightarrow \exists y'(\neg B(y',y') \wedge \theta_3(y,y'))).$$

Thus, the formula ψ_6 says if y is an element of the second component that represents a finite or cofinite set, then there is another element y' in the second component that represents a different set, although both y and y' encode the same real number. (We except from this the empty set and the set of all numbers, since they are the only sets encoding 0 and 1, respectively.) It follows that y' must represent the appropriate cofinite (resp. finite) set. The formulas ψ_5 and ψ_6 now guarantee that every set of natural numbers is represented by some element in the second component and every element in the second component represents some set of natural numbers.

In the Σ^2_1 sentence we wish to capture, there will also be existential quantification over sets of sets of natural numbers. Since the third-order quantifiers are existential and their number is finite, we have to represent a finite number of sets of sets, those needed for instantiating the third-order quantified variables. We represent these sets of sets of natural numbers by elements of the first component. As we mentioned above, we again use B to encode the membership relation. Thus, if y is in the second component and x is in the first component, we take $B(y,x)$ to hold if the set of natural numbers represented by y is an element of the set of sets represented by x. We cannot force there to be an element of the first component representing each possible set of sets of natural numbers; fortunately, we don't need to do this. (We would need to do this if we considered, say, Π^2_1 formulas.)

Naturally, we need to express that if y and y' represent the same set of natural numbers then they belong to the same sets of sets of natural numbers. This "extensionality" property is expressed by the following formula:

$$\begin{aligned}\psi_7 =_{\mathrm{def}} \quad \forall y, y', x(\neg B(y,y) \wedge \neg B(y',y') \wedge \\ \forall z(B(z,z) \Rightarrow B(z,y) \Leftrightarrow B(z,y')) \\ \Rightarrow (B(y,x) \Leftrightarrow B(y',x))).\end{aligned}$$

Notice that the ψ_2 above actually expressed a similar fact: that if x and x' represent the same number then they belong to the same sets of natural numbers.

We next provide a translation $\varphi \to \varphi^t$ from Σ^2_1 formulas to formulas in $\mathcal{L}_1(\{B\})$, by induction on the structure of φ, starting with atomic formulas. Without loss of generality, we can assume that all the atomic formulas have one of the forms: $x = 0$, $x = 1$, $x + x' = y$, $x \in X$, and $X \in \mathcal{X}$. (As we remarked earlier, we can assume without loss of generality that there is no multiplication in ψ; we can clearly rewrite ψ to get rid of more complicated atomic formulas such as $x + x' = y + y'$, etc.) In the translation, we treat all the variables $x, X, \mathcal{X}$ as object variables, and assume that the basic connectives in formulas are $\wedge$, $\neg$, and $\exists$.

- $(x = 0)^t \ = \ w_z(B(z,x)) = 1/2$
- $(x = 1)^t \ = \ w_z(B(z,x)) = 1/4$
- $(x + x' = y)^t \ = \ w_z(B(z,x)) \times w_z(B(z,x')) = \frac{1}{2} w_z(B(z,y))$
- $(x \in X)^t \ = \ B(x,x) \wedge \neg B(X,X) \wedge B(x,X)$
- $(X \in \mathcal{X})^t \ = \ \neg B(X,X) \wedge B(\mathcal{X},\mathcal{X}) \wedge B(X,\mathcal{X})$
- $(\varphi_1 \wedge \varphi_2)^t \ = \ \varphi_1^t \wedge \varphi_2^t$
- $(\neg\varphi)^t \ = \ \neg(\varphi^t)$
- $(\exists x \varphi)^t \ = \ \exists x(B(x,x) \wedge \varphi^t)$
- $(\exists X \varphi)^t \ = \ \exists X(\neg B(X,X) \wedge \varphi^t)$
- $(\exists \mathcal{X} \varphi)^t \ = \ \exists \mathcal{X}(B(\mathcal{X},\mathcal{X}) \wedge \varphi^t)$

Finally, let ψ' be the conjunction of ψ^t and the formulas $\psi_1, \ldots, \psi_7$. We claim that ψ is true iff ψ^t is satisfiable.

This argument shows that the satisfiability problem for $\mathcal{L}_1(B)$ is Σ_1^2 hard, and hence that the validity problem is Π_1^2 hard.

Next, we discuss the Π_1^2 upper bound. We prove that the satisfiability problem is Σ_1^2, using two basic ideas. The first idea is that we need to consider only models which are not "too large"—at most the cardinality of the continuum; we exploit the discreteness of the probability distribution to show this. The second idea is that, with a little higher-order quantification, we can define when a function is a probability function and when a formula holds with a given probability. Therefore, we can transform a formula in the logic of probability into a classical (higher-order) formula. Thus, the satisfiability problem is reduced to a classical one. We sketch the proof here, leaving many details to the reader.

Consider a sentence φ in $\mathcal{L}_1(\Phi)$, Without loss of generality, we assume that the probability operator occurs only in formulas of the form $w_{\vec{y}}(\varphi) = x^f$. Our first step is to transform φ to a formula φ' in a richer one-sorted language that does not involve probability terms. In the richer language we have new unary predicate symbols isD, isR, isN, functions $h_1, \ldots, h_k$, $\mu_1, \ldots, \mu_k$ from the natural numbers to the reals, and binary predicates $Elem_1, \ldots,$

$Elem_k$ on the natural numbers, where k is the length of the longest vector $\vec{x}$ that occurs in a subterm $w_{\vec{x}}(\varphi')$ of φ. Intuitively, isD is true for elements that are meant to represent elements of the object domain, isR is true for the reals, and isN is true for the natural numbers (which, of course, are intended to be a subset of the reals). The object domain is no longer assumed to be disjoint from the reals, and in fact will be a subset of the reals in the structure constructed in this proof. We explain the role of the h_i's below.

Transforming φ to a formula in a one-sorted language uses standard techniques from logic; we simply relativize quantification over the object domain and reals by using isD and isR. Further transforming φ so that it does not involve probability terms requires more effort. Here is where the functions h_i, μ_i and the predicates $Elem_i$ come in. Since there are only countably many elements in the domain that can have positive probability (thanks to our assumption that the probability is discrete), we essentially assume that these elements are in fact all natural numbers. (In particular, we do not assume that isD and isN represent disjoint sets.) We then force μ_i to assign probability to i-tuples and h_i to assign probability to finite sets of i-tuples in a consistent way. We discuss the case of h_1 and μ_1 here—the other ones follow a similar pattern.

Although h_1 is a function from the natural numbers to the reals, we really want to view it as a function from finite sets of natural numbers to the reals. Intuitively, $h_1(n)$ is the probability of the set represented by n. There are many ways to encode a finite set of natural numbers as a natural number. We use the following. Let $\tau(m,n) = \frac{1}{2}(m^2 + 2mn + n^2 + 3m + n)$. As observed in [Rog67], τ is a recursive one-one mapping of $\mathbb{N} \times \mathbb{N}$ onto $\mathbb{N}$. We take 0 to be an encoding of the empty set, and take $\tau(m,n)$ to be the union of $\{m\}$ and the set encoded by n.

We take $Elem_1(m,n)$ to hold exactly if m is a member of the set represented by n under this encoding. The following two formulas force $Elem_1$ to have the required properties:

- $\forall m \neg Elem_1(m,0)$
- $\forall m,n[Elem_1(m,n) \Leftrightarrow$
  $isN(m) \wedge isN(n) \wedge$
  $\exists k_1,k_2(isN(k_1) \wedge isN(k_2) \wedge n = \tau(k_1,k_2) \wedge (m = k_1 \vee Elem_1(m,k_2))]$

The following four formulas guarantee that h_1 and μ_1 "work right". The first one guarantees that the probability of the empty set is 0, while the second one deals with adding an element to a set. The other two guarantee that probabilities sum up to 1.

- $h_1(0) = 0$
- $\forall m, n(isN(m) \wedge isN(n) \wedge \neg Elem_1(m, n)$ $\Rightarrow h_1(\tau(m, n)) = \mu_1(m) + h_1(n))$
- $\forall m, n(isN(m) \wedge isN(n) \wedge Elem_1(m, n) \Rightarrow h_1(\tau(m, n)) = h_1(n))$
- $\forall k(isN(k) \Rightarrow \exists n(isN(n) \wedge h_1(n) \geq 1 - 1/k))$
- $\forall n(isN(n) \Rightarrow h_1(n) \leq 1 \wedge \mu_1(n) \geq 0)$

With all these definitions in hand, it is now easy to replace all occurrences of formulas such as $w_{\vec{y}}(\psi) = r$. Suppose the $\vec{y}$ actually consists of just the variable y. We then replace the formula $w_y(\psi) = r$ by:

$$\begin{aligned}\forall k \exists n_1, n_2 \forall m_1, m_2((Elem_1(m_1, n_1) \Rightarrow \psi(m_1)) \wedge \\ (Elem_1(m_2, n_2) \Rightarrow \neg\psi(m_2)) \wedge \\ h_1(n_1) > r - 1/k \;\wedge\; h_1(n_2) > 1 - r - 1/k).\end{aligned}$$

Intuitively, this formula says we can find finite subsets n_1 and n_2 of the set of elements that satisfy ψ and $\neg\psi$, respectively, such that the probability of n_1 is arbitrarily close to r and the probability of n_2 is arbitrarily close to $1 - r$. Similar ideas work (using h_i and $Elem_i$) if $\vec{y}$ is a vector of length i.

Let φ' be the formula that results by transforming φ as described above and conjoining the axioms that describe $h_1, \ldots, h_k, \mu_1, \ldots, \mu_k, Elem_1, \ldots, Elem_k$. We leave it to the reader to check that φ is satisfiable iff φ' is satisfiable in an *appropriate* structure: one that interprets isR as the reals, isN as the natural numbers (a subset of the reals), and interprets 0, 1, +, and < in the standard way over the reals. Notice that if we can find an appropriate structure M satisfying φ', we can take the Skolem hull of $I\!R$ [CK90] in M to find an appropriate structure M' satisfying φ' such that M' has the cardinality of $I\!R$. It is now straightforward to prove that φ' is satisfiable in an appropriate structure iff φ' is satisfiable in an appropriate structure with domain $I\!R$, where isD is interpreted as a subset of $I\!R$. (The proof is by induction on the structure of φ', using the fact that isD only occurs when it is necessary to relativize quantifications.)

It follows that the original formula φ is satisfiable if and only if there exist relations and functions $B_1, \ldots, B_n$, isD over the reals, functions $h_1, \ldots, h_k$, $\mu_1, \ldots, \mu_k$ from the natural numbers to the reals, and relations $Elem_1, \ldots,$

$Elem_k$ on the natural numbers such that φ' holds. (The predicate isR is no longer needed, as it can be taken to be identically true.) Thus, φ is satisfiable if and only if $\exists B_1, \ldots, B_n, isD, h_1, \ldots, h_k, \mu_1, \ldots, \mu_k, Elem_1, \ldots, Elem_k \varphi'$ is true. Here all higher-order quantifiers range over operations on reals; after this prefix, φ' itself contains only first-order quantifiers over the reals.

We conclude that the satisfiability problem is in Σ_1^2, and hence that the validity problem is in Π_1^2. ∎

Up to now we have assumed that equality is not in the language (that is, we considered $\mathcal{L}_1(\Phi)$, not $\mathcal{L}_1^=(\Phi)$). For the classical predicate calculus, it is known that if we have only unary predicates and equality, then the validity problem is still decidable [DG79]. In this case, we might expect an analogue of Theorem 5.1. Unfortunately, once we introduce equality, we get bad undecidability without any predicates at all! More precisely, we prove:

Theorem 5.3: *For all Φ (even if Φ is empty) the validity problem for $\mathcal{L}_1^=(\Phi)$ is Π_∞^1 hard. If Φ contains at most unary predicate symbols and constant symbols, then the validity problem for $\mathcal{L}_1^=(\Phi)$ is Π_∞^1 complete. If Φ has a binary predicate, then the validity problem for $\mathcal{L}_1^=(\Phi)$ is Π_1^2 complete.*

Proof: The Π_1^2 lower bound with a binary predicate in Φ follows from Theorem 5.2. In the case that Φ is empty, we prove the Π_∞^1 lower bound by showing that we can define a predicate isN such that $isN(r)$ holds exactly if r is a natural number. We do this by using a slightly different encoding of natural numbers than that used in Theorem 5.2. We encode the natural number n by a domain element with probability $1/(n+1)(n+2)$. Our first step is to force domain elements with this probability to exist. This is the job of ψ_1 and ψ_2 below:

$$\begin{aligned}
\psi_1 =_{\text{def}} \;& \exists z[w_y(y = z) = 1/2];\\
\psi_2 =_{\text{def}} \;& \forall r \geq 0[(\exists x(r+1)(r+2)(w_y(y = x)) = 1)\\
& \qquad \Rightarrow (\exists x'(r+2)(r+3)(w_y(y = x')) = 1)].
\end{aligned}$$

In ψ_1 we think of z as 0, and in ψ_2 we think of x' as the successor of x. Since

$$\sum_{n=0}^{\infty} \frac{1}{(n+1)(n+2)} = 1,$$

the domain elements with positive probability, equipped with this 0 and this successor relation, are isomorphic to the standard natural numbers. Furthermore, the only possible positive probabilities for domain elements are of the form $1/(n+1)(n+2)$, where n is a natural number. Thus, we can pick out the reals which are natural numbers: these are exactly the r such that

$$\exists x[(r+1)(r+2)(w_y(y=x))=1].$$

This immediately gives us a predicate isN to test for whether a number is a natural number. It is now straightforward to translate a formula φ in the language of elementary analysis to an equisatisfiable formula φ' of $\mathcal{L}_1^=(\emptyset)$. We simply replace quantification over the natural numbers by quantification over the reals, relativized to these reals that satisfy isN. This yields the Π^1_∞ lower bound.

The Π^2_1 upper bound follows from the proof of Theorem 5.2 (adding equality to the language does not affect the proof at all). We now show that we can reduce the complexity to Π^1_∞ if we have at most unary predicates in the languages, matching the lower bound proved above. The proof is similar to that of Theorem 5.2; we just need one additional observation, which allows us to restrict attention to *countable structures*, that is, structures with countable domains.

Lemma 5.4: *If Φ consists only of unary predicates symbols and constant symbols, then a formula φ in $\mathcal{L}_1^=(\Phi)$ is satisfiable iff φ is satisfiable in a countable structure.*

Proof: Suppose φ is a satisfiable formula in $\mathcal{L}_1^=(\Phi)$. Suppose that it is satisfied in some structure M. We show that φ is in fact satisfied in some countable substructure M' of M. The proof uses similar techniques to that showing that if a formula of first-order logic with only unary predicates is satisfied in some structure M, then it is satisfied in a finite substructure of M [DG79]. Let $P_1(x),\ldots,P_m(x)$ be the unary predicates in φ that appear in φ, and let $a_1,\ldots,a_n$ be the constant symbols that appear in φ. Let an *atom* over $P_1,\ldots,P_m$ be a formula of the form $Q_1(x)\wedge\ldots\wedge Q_m(x)$, where each Q_i is either P_i or $\neg P_i$. Notice that there are 2^m such atoms; call them $A_1(x),\ldots,A_{2^m}(x)$. The atoms partition the elements of D into equivalence classes $D_1,\ldots,D_{2^m}$, where D_i consists of all the elements of D that satisfy atom A_i. Let D'' consist of all the domain elements with positive probability and the domain elements which are the interpretations of the constant symbols $a_1,\ldots,a_n$. Let D' consist of D'' together with all of D_i if D_i is countable, or a countably infinite subset D_i' of D_i if D_i is not countable, for $i=1,\ldots,2^m$. By construction, D' is countable. Furthermore, we may assume that D' is infinite, since if D' is finite then so is D, and then the lemma is proved.

Let $\Phi' = \{P_1, \ldots, P_m, a_1, \ldots, a_n\}$ and let M' be the obvious restriction of M to D' for the symbols in Φ'. (The interpretation in M' of the symbols in $\Phi - \Phi'$ is irrelevant.)

Claim 1: For all formulas ψ and real terms t in $\mathcal{L}_1^{=}(\Phi')$, if v is a valuation that maps all variables into elements of D', then $M, v \models \psi$ iff $M', v \models \psi$ and $[t]_{(M,v)} = [t]_{(M',v)}$.

The proof of the claim proceeds by a straightforward induction on the structure of ψ and t. The only nontrivial case comes if ψ is of the form $\exists x \psi'$. It immediately follows from the induction hypothesis that if $(M', v) \models \exists x \psi'$ then $(M, v) \models \exists x \psi'$. For the converse, we require the following "automorphism property":

Claim 2: If τ is an automorphism of D that keeps D'' fixed and respects D_i, $i = 1, \ldots, 2^m$ (that is, if $d \in D_i$ then $\tau(d) \in D_i$, $i = 1, \ldots, 2^m$), then for all formulas ψ and real terms t in $\mathcal{L}_1^{=}(\Phi')$, we have $(M, \tau \circ v) \models \psi$ iff $(M, v) \models \psi$ and $[t]_{M,v} = [t]_{M,\tau\circ v}$.

The proof of this automorphism property again proceeds by induction on the structure of formulas and terms. Again, the only difficulty comes if ψ is of the form $\exists x \psi'$. In this case, we have

$$\begin{array}{ll} & (M, v) \models \exists x \psi' \\ \text{iff} & (M, v[x/d]) \models \psi' \text{ for some } d \in D \\ \text{iff} & (M, \tau \circ (v[x/d])) \models \psi' \text{ for some } d \in D \text{ (by the induction hypothesis)} \\ \text{iff} & (M, (\tau \circ v)[x/\tau(d)]) \models \psi' \text{ for some } d \in D \\ \text{iff} & (M, (\tau \circ v)[x/d]) \models \psi' \text{ for some } d \in D \text{ (since } \tau \text{ is an automorphism)} \\ \text{iff} & (M, \tau \circ v) \models \exists x \psi'. \end{array}$$

This proves Claim 2.

Returning to the proof of Claim 1, suppose that $(M, v) \models \exists x \psi'$. Then there exists $d \in D$ such that $(M, v[x/d]) \models \psi'$. If $d \in D'$ then, by the induction hypothesis, we have $(M', v[x/d]) \models \psi'$, and $(M', v) \models \exists x \psi'$. If $d \notin D'$, then let τ be an automorphism of D fixing D'' and respecting D_i, for $i = 1, \ldots, m$, such that $\tau(d) \in D'$, $\tau(D') \subseteq D'$, and $\tau(v(y)) = v(y)$ for free variables y of $\exists x \psi'$.

It is easy to check that such an automorphism exists, as follows. If $D = D'$, then we just take τ to be the identity. If $D \neq D'$, it must be the case that some D_i, and hence D, is uncountable and thus that $D - D'$ is (uncountably) infinite. Recall that (according to the hypotheses of Claim 1) $E = \{v(y) : y \text{ is a free variable of } \exists x \psi'\} \subseteq D'$. Choose some element $d' \in$

$D' - E$. Since $D' - E$ and $D - D'$ are infinite, there exists an automorphism τ such that τ is the identity on E, $\tau(d) = d'$, $\tau(D - D' - \{d\}) = D - D'$, and $\tau(D' - E) = D' - E - \{d'\}$. This gives us the required automorphism.

By Claim 2, we have $(M, \tau \circ (v[x/d])) \models \psi'$. Thus $(M, (\tau \circ v)[x/\tau(d)]) \models \psi'$. Since $(\tau \circ v)[x/\tau(d)]$ is a valuation which maps all variables to D', by the induction hypothesis we have $(M', (\tau \circ v)[x/\tau(d)]) \models \psi'$, so $(M', \tau \circ v) \models \exists x \psi'$. Since $\tau \circ v$ and v agree on all the variables free in ψ', we must also have $(M', v) \models \exists x \psi'$. This completes the proof of Claim 1 and of the lemma. ∎

Thus, we have proved that we can restrict attention to countable structures. We can now use the techniques of the upper-bound proof for the general case, to show that the original formula φ is satisfiable if and only if there exist predicates isD, P_1, ..., P_m over the naturals, and functions μ_1, ..., μ_k, h_1, ..., h_k from the naturals to the reals, and predicates $Elem_1$, ..., $Elem_k$ on the natural numbers such that the formula φ' holds. Here all second-order quantifiers range over operations on naturals; after this prefix, φ' itself contains only first-order quantifiers over naturals and reals.

We conclude that the satisfiability problem is in Σ^1_∞, and hence that the validity problem is in Π^1_∞. ∎

As we remarked in the introduction, Bacchus gives a complete axiomatization for a logic syntactically similar to ours, thus showing that the validity problem is semi-decidable in his semantics. The reason for this difference is that Bacchus allows nonstandard probability functions, which are required only to be finitely additive and can take values in arbitrary ordered fields. The proofs above require only the probability function to be finitely additive. Thus, the key reason that Bacchus is able to obtain a complete axiomatization is that he allows probabilities to take values in arbitrary ordered fields. As observed by Gaifman, Bacchus' result can perhaps best be understood by observing that ordered fields can be characterized by a finite collection of first-order axioms, as can finite additivity (while countable additivity cannot). Using this observation, it can be shown that Bacchus' language can be translated into first-order logic. From this, axiomatizability follows from the axiomatizability of first-order logic.

Validity is intractable for real-valued probabilities, as we have shown; it is also intractable for other fields, such as the rationals:

Theorem 5.5: *If the underlying field is that of the rational numbers, then*

(a) for all Φ the validity problem for $\mathcal{L}_1^=(\Phi)$ is Π^1_1 complete;

(b) *if Φ contains at least a binary predicate, then the validity problem for $\mathcal{L}_1(\Phi)$ is Π^1_1 complete; and*

(c) *if Φ consists of only unary predicates, then the validity problem for $\mathcal{L}_1(\Phi)$ is Π^0_∞ complete.*

Proof: In order to prove Theorem 5.5, we need to review some material from [Hal91]. Recall that the language for Presburger arithmetic consists of the constants 0, 1, and the function symbol $+$, interpreted over the natural numbers. Thus, Presburger arithmetic is the theory of arithmetic without multiplication. The validity problem of Presburger arithmetic is well known to be decidable [End72]. However, once we add a free unary predicate P to the language, so that we can write formulas such as $\forall x \forall y \forall z(P(x) \wedge P(y) \wedge x + y = z \Rightarrow P(z))$, the situation becomes drastically different. Garfunkel and Schmerl proved that Presburger arithmetic with a unary predicate is undecidable [GS74]; in fact, the following stronger result can be shown:

Theorem 5.6: [Hal91] *The validity problem for Presburger arithmetic with one unary predicate is Π^1_1 complete.*

In order to prove (a), we translate formulas of Presburger arithmetic with one unary predicate into $\mathcal{L}_1^=(\emptyset)$. Much as in Theorem 5.2, we can encode the natural number n by a domain element with probability $1/2^{n+1}$. In order to get the Π^1_1 lower bound, however, we need to be able to represent an additional unary predicate P over natural numbers. For this purpose, we use a slightly more complicated encoding of the natural number n as a pair of domain elements, whose probability is approximately equal and sums to $1/2^{n+1}$.

More precisely, let $\rho(x_1, x_2)$ be a formula that expresses that x_1 and x_2 are different domain elements but nearer in probability to each other than to any other domain element:

$$\begin{aligned}\rho(x_1, x_2) =_{\rm def} \quad & x_1 \neq x_2 \wedge \\ & \forall y(y \neq x_1 \wedge y \neq x_2 \Rightarrow \\ & |w_z(z = x_1) - w_z(z = x_2)| < |w_z(z = x_1) - w_z(z = y)| \wedge \\ & |w_z(z = x_1) - w_z(z = x_2)| < |w_z(z = x_2) - w_z(z = y)|).\end{aligned}$$

(Clearly absolute value is expressible in our language.) The formulas ψ_1 and ψ_2 below guarantee that we have pairs with total probability 1/2, 1/4, 1/8,

.... We use the pair with probability $1/2^{n+1}$ to encode the natural number n.

$$\begin{aligned}
\psi_1 =_{\rm def} &\ \exists x_1, x_2(\rho(x_1,x_2) \wedge (w_y(y=x_1) + w_y(y=x_2) = 1/2)) \\
\psi_2 =_{\rm def} &\ \forall r[\exists x_1, x_2(\rho(x_1,x_2) \wedge (w_y(y=x_1) + w_y(y=x_2) = r)) \\
&\quad \Rightarrow \exists x_1', x_2'(\rho(x_1',x_2') \wedge (w_y(y=x_1') + w_y(y=x_2') = r/2))].
\end{aligned}$$

The following formula guarantees that every domain element has positive probability; this gives us a straightforward bijection between pairs (d, d') of domain elements satisfying ρ and the natural numbers:

$$\psi_3 =_{\rm def} \forall x(w_z(z=x) > 0).$$

We next provide a translation $\varphi \rightarrow \varphi^t$ from a formula φ of Presburger arithmetic with one unary predicate P to a formula φ^t of $\mathcal{L}_1^{=}(\emptyset)$. Without loss of generality, we may consider only formulas φ all of whose atomic subformulas are of the form $P(x)$, $x = 0$, $x = 1$, and $x + x' = y$. Corresponding to each variable x that appears in φ, we have the pair of variables x_1, x_2 in φ^t. The key trick in the translation is that we encode $P(n)$ by taking the pair of elements that encode n to have equal probability, while we encode $\neg P(n)$ by taking the pair of elements that encode $P(n)$ to have distinct probabilities. The translation proceeds as follows. (The reader should compare this translation with the one used in Theorem 5.2.)

- $(x=0)^t \;=\; (w_z(z=x_1) + w_z(z=x_2) = 1/2)$
- $(x=1)^t \;=\; (w_z(z=x_1) + w_z(z=x_2) = 1/4)$
- $(x+x'=y)^t \;=$ $(w_z(z=x_1) + w_z(z=x_2)) \times (w_z(z=x_1') + w_z(z=x_2')) \;=$ $\frac{1}{2}(w_z(z=y_1) + (w_z(z=y_2))$
- $(P(x))^t \;=\; (w_z(z=x_1) = w_z(z=x_2))$
- $(\varphi_1 \wedge \varphi_2)^t \;=\; \varphi_1^t \wedge \varphi_2^t$
- $(\neg\varphi)^t \;=\; \neg(\varphi^t)$
- $(\exists x \varphi)^t \;=\; \exists x_1, x_2(\rho(x_1,x_2) \wedge \varphi^t)$

Finally, given a sentence ψ of Presburger arithmetic with an additional unary predicate, let $\psi' =_{\rm def} \psi_1 \wedge \psi_2 \wedge \psi_3 \wedge \psi^t$. It is now easy to check that ψ is satisfiable (over $\mathbb{N}$) iff ψ' is satisfiable with probabilities over the rationals. This gives the required lower bound for part (a) in Theorem 5.5.

The upper bound for (a) can be proved as in Theorem 5.2, replacing the reals with the rationals. A drop from Π_1^2 to Π_1^1 accompanies this drop in cardinality, since now all the predicates and functions are now defined over the rationals (and hence can be viewed as being defined over the natural numbers), rather than the reals.

The proof of (b) is essentially identical to that of (a). Suppose there is a binary predicate B in Φ. We can force B to be an equivalence relation, and then replace all occurrences of subformulas of the form $x = y$ by $B(x, y)$ in the translation from ψ to ψ'. We leave details to the reader.

Finally, for (c) observe that we get a Π_0^∞ lower bound since the theory of the rationals with $+$ and $\times$ is already arithmetic (that is, Π_0^∞-complete) by a result of Robinson [Rob49]. (The key step in Robinson's proof is showing that a predicate testing whether a number is a natural number can be defined in this language.) For the upper bound, we use techniques similar to those of Theorem 5.1. Indeed, the proof of Claim 2 of Theorem 5.7 in [Hal90] shows that, given a formula φ in $\mathcal{L}_1(\Phi)$, where Φ consists of only unary predicates, we can find first-order formulas $\varphi_1, \ldots, \varphi_k$ and mutually exclusive formulas $\psi_1, \ldots, \psi_k$ in the language of real closed fields such that φ is valid (given that probabilities take only rational values) iff $(\varphi_1 \wedge \psi_1) \vee \ldots \vee (\varphi_k \wedge \psi_k)$ is valid (where $\psi_1, \ldots, \psi_k$ are interpreted over the rationals). Since the validity of $(\varphi_1 \wedge \psi_1) \vee \ldots \vee (\varphi_k \wedge \psi_k)$ can easily be encoded by a formula of arithmetic, the upper bound follows. ∎

The situation is analogous and worse when we move to type 2 structures. Here we get to Π_1^2 completeness as soon as there is even one unary predicate in the language. If equality is included as a logical symbol, then we get Π_∞^1 completeness with only a constant symbol in Φ. (If φ does not contain any nonlogical symbols, then $\varphi \Rightarrow (w(\varphi) = 1)$ is valid; thus we cannot make any nontrivial probability statements if Φ is empty.)

Theorem 5.7: *If Φ contains at least one predicate of arity greater than or equal to one then the validity problem for $\mathcal{L}_2(\Phi)$ with respect to type 2 probability structures is Π_1^2 complete.*

Proof: The lower-bound proof has a similar flavor to that of Theorem 5.2, so we only sketch the highlights here. Suppose Φ contains the unary predicate P. Given a Σ_1^2 formula ψ, we want to construct effectively a formula ψ' in

$\mathcal{L}_2(\{P\})$ such that ψ is satisfiable iff ψ' is satisfiable. As we mentioned in Section 4, in order to prove the Σ_1^2 lower bound, it suffices to provide such an effective translation for formulas ψ with at most one existential third-order quantifier.

As in the proof of Theorem 5.2, we think of the domain as consisting of two components. Fix a state s; the type 1 elements are those for which $(M,s) \models P(d)$, while the type 2 elements are those for which $(M,s) \models \neg P(d)$. We associate with every domain element d the set S_d of states of positive probability such that $P(d)$ holds. We construct formulas $\psi_1, \ldots, \psi_4$ such that if $(M,s) \models \psi_1 \wedge \ldots \wedge \psi_4$, then for all type 1 domain elements d, it must be the case that S_d has probability one of $1/2, 1/4, 1/8, \ldots$. Intuitively, we take d to represent the natural number n if the probability of the set S_d is $1/2^{n+2}$. Domain elements d such that the probability of S_d is $1/2$ will be used to encode a set of sets of natural numbers. (It will suffice to encode a single set of sets since we are considering only formulas with a single third-order existential quantifier.) The type 2 domain elements will be used to encode sets of natural numbers.

We proceed as follows. The formula ψ_1 says that for each type 1 element d, the set S_d is nonempty (so that the set of states where $P(d)$ holds has positive probability). The formula ψ_2 says that there exists a type 1 element d such that S_d has probability $1/2$. The formula ψ_3 guarantees the existence of type 1 elements d_i such that the probability of S_{d_i} is $1/2^i$. Finally, ψ_4 guarantees that the sets of states with probability $1/2^k$ are all *almost disjoint*, that is, their intersection has measure 0. More precisely, we show that if $\mu(S_d) \neq \mu(S_{d'})$, then $\mu(S_d \cap S_{d'}) = 0$. As a consequence of this, we can show that there exist elements $d_1, d_2, \ldots$ such that $\mu(S_{d_i}) = 1/2^i$, $\mu(\cup_i S_{d_i}) = 1$, and $\mu(S_{d_i} \cap S_{d_j}) = 0$ if $i \neq j$. It follows that, for any element d in the domain, if $\mu(S_d) = 1/2^i$, then $\mu(S_d \cap S_{d_i}) = 1/2^i$. Thus, if $\mu(S_d) = 1/2^i$, then S_d and S_{d_i} are *almost identical*; their set difference has measure 0.

$$\begin{aligned}
\psi_1 &=_{\text{def}} \forall x(P(x) \Rightarrow w(P(x)) > 0)\\
\psi_2 &=_{\text{def}} \exists x(P(x) \wedge w(P(x)) = 1/2)\\
\psi_3 &=_{\text{def}} \forall x(P(x) \Rightarrow \exists y(P(y) \wedge 2w(P(y)) = w(P(x))))\\
\psi_4 &=_{\text{def}} \forall x,y(P(x) \wedge P(y) \wedge w(P(x)) \neq w(P(y))\\
&\qquad\qquad \Rightarrow w(P(x) \wedge P(y)) = 0)
\end{aligned}$$

We next consider the relationship between type 1 and type 2 elements. The formula ψ_5 forces the set S_e for a type 2 domain element e to consist essentially of the union of sets S_d for type 1 domain elements d. (More precisely, S_e is almost identical to $\cup_{d \in A} S_d$, for some set A of type 1 elements.)

$$\begin{array}{ll} \psi_5 =_{\rm def} & \forall x, y(P(x) \wedge \neg P(y) \\ & \qquad \Rightarrow w(P(x) \wedge P(y)) = 0 \vee w(P(x) \wedge P(y)) = w(P(x))). \end{array}$$

Informally, we can associate every type 2 element e with a binary decimal, just as in the proof of Theorem 5.2. However, in this case, we plan to use the leading bit of the decimal to tell us whether the set of natural numbers that e represents is in the set of sets that we are encoding, while the remaining elements of the decimal describe the elements of the set that e represents. Thus, the number n is in the set represented by e if there is a type 1 element d such that the probability of S_d is $1/2^{n+2}$ (thus, d represents n) and S_d is essentially a subset S_e (that is, $S_d - S_e$ has measure 0). The set represented by e is in the set of sets we are encoding if there is a type 1 element d such that the probability of S_d is $1/2$ and S_d is essentially a subset of S_e.

The formula ψ_6 is the analogue of ψ_5 in the proof of Theorem 5.2. It guarantees that, for every real number r between 0 and $1/2$, there is a type 2 domain element e such that the probability of S_e is either r or $r + 1/2$. The complication here arises because of our use of domain elements with probability $1/2$ to encode a set of sets. Just as in the proof of Theorem 5.2, the element e represents a set of natural numbers; this set is in the set of sets encoded by (the type 1 elements of probability $1/2$ in) the structure iff the probability of e is $r + 1/2$.

$$\psi_6 =_{\rm def} \forall r(0 \leq r \leq 1/2 \Rightarrow \exists y(\neg P(y) \wedge (w(P(y)) = r \vee w(P(y)) = r + 1/2)).$$

The formula ψ_7 is an analogue of ψ_6 in Theorem 5.2; it takes care of the case where a real number encodes two possible sets of natural numbers. This will guarantee that every set of natural numbers is represented by some domain element of type 2. First we need formulas θ_1, θ_2, and θ_3 that are analogues of the formulas with the same names used in Theorem 5.2:

$$\begin{array}{ll} \theta_1(y) =_{\rm def} & \exists x(P(x) \wedge w(P(x)) < 1/2 \wedge w(P(x) \wedge P(y)) > 0) \wedge \\ & \exists r > 0(\forall x(P(x) \wedge w(P(x)) < r \Rightarrow w(P(x) \wedge P(y)) = 0)) \\ \theta_2(y) =_{\rm def} & \exists x(P(x) \wedge w(P(x)) < 1/2 \wedge w(P(x) \wedge P(y)) = 0) \wedge \\ & \exists r > 0(\forall x(P(x) \wedge w(P(x)) < r \Rightarrow w(P(x) \wedge P(y)) > 0)) \\ \theta_3(y, y') =_{\rm def} & (w(P(y)) = w(P(y')) \vee |w(P(y)) - w(P(y'))| = 1/2) \wedge \\ & \exists x'(P(x') \wedge w(P(x')) \neq 1/2 \wedge \\ & \qquad w(P(x') \wedge P(y)) > 0 \wedge w(P(x') \wedge P(y')) = 0)) \end{array}$$

Then we obtain ψ_7:

$$\psi_7 =_{\text{def}} \quad \forall y(\neg P(y) \wedge (\theta_1(y) \vee \theta_2(y)) \Rightarrow \exists y'(\neg P(y') \wedge \theta_3(y, y'))).$$

Just as in Theorem 5.2, we need to guarantee that if two distinct type 2 elements e and e' represent the same set, then either both belong to the set of sets or neither does. This is the job of the following formula ψ_8, which is an analogue of ψ_7 in Theorem 5.2:

$$\begin{aligned}\psi_8 =_{\text{def}} \quad & \forall y, y'(\neg P(y) \wedge \neg P(y') \wedge \\ & \qquad \forall x(P(x) \wedge w(P(x)) < 1/2 \\ & \qquad\quad \Rightarrow w(P(x) \wedge P(y)) = w(P(x) \wedge P(y'))) \\ & \qquad \Rightarrow (w(P(y)) \geq 1/2) \Leftrightarrow (w(P(y')) \geq 1/2)).\end{aligned}$$

After these preliminaries, we are ready to provide a translation $\varphi \to \varphi^t$ from a Σ_1^2 formula φ to a $\mathcal{L}_2(\{P\})$ formula φ^t by induction on the structure of φ, starting with atomic formulas. In the translation, we treat all the variables $x, X, \mathcal{X}$ as object variables, and assume that the basic connectives in formulas are $\wedge$, $\neg$, and $\exists$.

- $(x = 0)^t \;=\; [w(P(x)) = 1/4]$
- $(x = 1)^t \;=\; [w(P(x)) = 1/8]$
- $(x + x' = y)^t \;=\; [w(P(x)) \times w(P(x')) = \frac{1}{4}w(P(y))]$
- $(x \in X)^t \;=\; P(x) \wedge \neg P(X) \wedge [w(P(x) \wedge P(X)) > 0]$
- $(X \in \mathcal{X})^t \;=\; \neg P(X) \wedge P(\mathcal{X}) \wedge [w(P(X) \wedge P(\mathcal{X})) > 0]$
- $(\varphi_1 \wedge \varphi_2)^t \;=\; \varphi_1^t \wedge \varphi_2^t$
- $(\neg\varphi)^t \;=\; \neg(\varphi^t)$
- $(\exists x\varphi)^t \;=\; \exists x(P(x) \wedge w(P(x)) \neq 1/2 \wedge \varphi^t)$
- $(\exists X\varphi)^t \;=\; \exists X(\neg P(X) \wedge \varphi^t)$
- $(\exists \mathcal{X}\varphi)^t \;=\; \exists \mathcal{X}(P(\mathcal{X}) \wedge w(P(\mathcal{X})) = 1/2 \wedge \varphi^t)$

Finally, let ψ' be the conjunction of ψ^t and the formulas $\psi_1, \ldots, \psi_8$. We leave it to the reader to check that ψ is true iff ψ^t is satisfiable.

The upper bound is proved in essentially the same way as that of Theorem 5.2. We remark that we can get an alternative proof of the upper bound from the upper bound for $\mathcal{L}_1^{=}(\Phi)$ proved in Theorem 5.2 together with the translation given in Theorem 6.2 from $\mathcal{L}_2$ formulas to equisatisfiable formulas in $\mathcal{L}_1$. We leave details to the reader. ∎

Theorem 5.8: *If Φ is nonempty then the validity problem for $\mathcal{L}_2^{=}(\Phi)$ with respect to type 2 probability structures is Π^1_∞ hard. If Φ contains at most constant symbols then the problem is Π^1_∞ complete.*

Proof: It clearly suffices to prove the lower bound for the case where Φ consists of one constant symbol, say c. In this case, the argument is almost identical to the hardness argument in the proof of Theorem 5.3. The only difference is that we replace formulas of the form $w_y(y = x)$ that appear in Theorem 5.3 by $w(c = x)$. We leave details to the reader.

The upper bound also resembles the upper bound in Theorem 5.3. Again, we restrict attention to the case where Φ contains only constant symbols, and we first show that we can restrict attention to countable structures.

Lemma 5.9: *If Φ consists only of constant symbols, then a formula in $\mathcal{L}_2^{=}(\Phi)$ is satisfiable iff it is satisfiable in a structure with a countable set of states and a countable domain.*

Proof: Suppose $(M, s, v) \models \varphi$, where $M = (S, D, \pi, \mu)$. Let S' consist of s and all the states $s' \in S$ such that $\mu(s') > 0$. Clearly S' is countable, since μ is a discrete measure. Let $M' = (S', D, \pi', \mu')$, where π' and μ' are the obvious restriction of π and μ to S'. It is easy to see that for all formulas ψ, all states $s' \in S'$, and all valuations v, we have $(M, s', v') \models \psi$ iff $(M', s', v') \models \psi$. In particular, it follows that $(M', s, v) \models \varphi$, so φ is satisfiable in a structure with a countable state space.

To see that we can further reduce to a countable domain, we proceed much as in the proof of Lemma 5.4. Suppose $a_1, \ldots, a_k$ are the constant symbols appearing in φ. Let D_0 consist of all the domain elements of the form $\pi(s')(a_i)$ for $s' \in S'$ and $i = 1, \ldots, k$. Clearly D_0 is countable, since S' is. If $D - D_0$ is finite, then D is countable, and we are done. Otherwise, let D' consist of D_0 together with a countably infinite subset of $D - D_0$. (The particular choice of subset is irrelevant.) Let $M'' = (S', D', \pi', \mu')$. It is now easy to prove the analogues of the two claims in Lemma 5.4, namely, taking $\Phi' = \{a_1, \ldots, a_k\}$, we can prove:

Claim 3: For all formulas ψ and real terms t in $\mathcal{L}_2^{=}(\Phi')$ and all $s' \in S'$, if v is a valuation that maps all variables into elements of D', then $(M', s', v) \models \psi$ iff $(M'', s', v) \models \psi$ and $[t]_{(M', s', v)} = [t']_{(M'', s', v)}$.

Again, the proof of the claim proceeds by a straightforward induction on the structure of ψ and t, with the only difficulty coming if ψ is of the form $\exists x \psi'$. To take care of this case, we prove another "automorphism property":
Claim 4: If τ is an automorphism of D' that keeps D_0 fixed, then for all formulas ψ and real terms t in $\mathcal{L}_2^=(\Phi')$, all states $s' \in S'$, and all valuations with range D', we have $(M', s', \tau \circ v) \models \psi$ iff $(M', s', v) \models \psi$ and $[t]_{M',s',v} = [t]_{M,s',\tau \circ v}$.

The proof of Claims 3 and 4 is essentially identical to that of Claims 1 and 2, so we leave details to the reader.

Thus, we have proved that we can restrict attention to countable structures and the upper bound follows, again using techniques of Theorem 5.2. ∎

The situation may not be quite as bleak as it looks. In many contexts, the domain is known to be of bounded size. If we restrict attention to structures of size at most N (for some fixed N), then we do get decidability.

Theorem 5.10: *For all Φ, the validity problem for $\mathcal{L}_1^=(\Phi)$ (resp. $\mathcal{L}_2^=(\Phi)$) with respect to type 1 (resp. type 2) probability structures of size at most N is decidable.*

Proof: In the proof of Theorem 5.8 (resp. Theorem 5.10) in [Hal90], it is shown that given a formula φ in $\mathcal{L}_1^=(\Phi)$ (resp. in $\mathcal{L}_2^=(\Phi)$) we can find formulas $\varphi_1, \ldots, \varphi_N$, ψ, such that:

- $\varphi_1, \ldots, \varphi_N$ are pure first-order formulas over Φ,
- ψ is a formula in the language of real closed fields, and
- φ is valid in structures of size at most N iff ψ is valid and φ_i is valid in structures of size i.

The result now follows from the decidability of the theory of real closed fields and the decidability of first-order logic in domains of fixed finite size. ∎

A fortiori, the same result holds when equality is not in the language. We can also get decidability if we restrict attention to structures of size exactly N for some fixed N.

On the other hand, the restriction to bounded structures is necessary.

Theorem 5.11: *For all Φ (resp., for all nonempty Φ) the validity problem for $\mathcal{L}_1^=(\Phi)$ (resp. $\mathcal{L}_2^=(\Phi)$) with respect to type 1 (resp. type 2) probability structures of finite size is co-r.e. complete.*

Proof: Since, by Theorem 5.10, the validity problem, and hence the satisfiability problem, is decidable for structures of size at most N, it is easy to see that the satisfiability problem is r.e. for finite domains, and hence the validity problem is co-r.e. If we have at least a binary predicate in Φ, then the fact that the validity problem for finite structures is co-r.e. hard follows from the well-known result that it is already co-r.e. hard for first-order logic [Tra50]. (This is true even without equality in the language.) To see that the lower-bound result holds even for $\mathcal{L}_1^=(\emptyset)$, we use similar ideas to those used in the proof of Theorem 5.5 to show that for every formula φ in the language of Presburger arithmetic augmented by a unary predicate P, we can find a formula φ^t of $\mathcal{L}_1^=(\emptyset)$ such that φ^t is valid in finite structures iff φ is valid when interpreted over finite initial segments of the natural numbers. (That is, we interpret these formulas over the domain $\{1, \ldots, N\}$, and $m + n = N$ holds in this interpretation if $m + n > N$ holds over the natural numbers.) A straightforward modification of the proof given in [Hal91] can be used to show that the validity problem for formulas in the language of Presburger arithmetic interpreted over finite initial segments of the natural numbers is co-r.e. complete; we omit details here. The lower bound in the case of $\mathcal{L}_2^=(\Phi)$ for nonempty Φ follows similar lines. For example, if Φ contains the constant symbol c, we would use formulas of the form $w(c = x)$ rather than formulas of the form $w_y(y = x)$. ∎

6 Translating between $\mathcal{L}_1$ and $\mathcal{L}_2$

In this section we show that in a precise sense the formalisms we have considered are equi-expressive. That is, we can effectively translate from $\mathcal{L}_1$ to $\mathcal{L}_2$ and from type 1 to type 2 structures, and vice versa, in a way that preserves satisfiability and validity of formulas. These results are somewhat surprising in light of the very different intuitions being captured by these two approaches. They also elucidate the relationship between the logics and their complexities.

We start with a translation from $\mathcal{L}_1$ to $\mathcal{L}_2$. We want to show that given a formula φ in $\mathcal{L}_1^=$ we can effectively find a formula φ^{12} in the language $\mathcal{L}_2^=$ and given a type 1 structure M we can construct a type 2 structure M^{12} such that $M \models \varphi$ iff $M^{12} \models \varphi^{12}$.

The idea is that we can replace subterms of φ of the form $w_{\vec{x}}(\psi)$ with $w(\psi[x_1/a_1, \ldots, x_n/a_n])$, where $a_1, \ldots, a_n$ are fresh constants, as long as

$a_1, \ldots, a_n$ act like "independent random variables". Formally, given a type 2 structure $M = (D, S, \pi, \mu)$, we say that $a_1, a_2, \ldots$ *are independent and identically distributed* in M if

- for all domain elements d and all i and j, we have

$$\mu(\{s : \pi(s)(a_i) = d\}) = \mu(\{s : \pi(s)(a_j) = d\})$$

- for all n, and all sequences $d_1, \ldots, d_n$ of elements in D, we have

$$\begin{aligned}&\mu(\{s : \pi(s)(a_1) = d_1, \ldots, \pi(s)(a_n) = d_n\}) = \\ &\quad \mu(\{s : \pi(s)(a_1) = d_1\} \times \cdots \times \mu(\{s : \pi(s)(a_n) = d_n\})\end{aligned}$$

Given a set Φ of predicate symbols and function symbols, let $\Phi^A = \Phi \cup \{a_1, a_2, \ldots\}$, where we assume that $a_1, a_2, \ldots$ are constant symbols not appearing in Φ. We say a type 2 structure is *special* with respect to Φ if $a_1, a_2, \ldots$ are independent and identically distributed in M, and each symbol in Φ has the same interpretation in all the states of M. As we now show, the special type 2 structures are in some sense equivalent to type 1 structures.

Theorem 6.1: *There is an effective translation that maps a formula φ in $\mathcal{L}_1^=(\Phi)$ to a formula φ^{12} in $\mathcal{L}_2^=(\Phi^A)$, and a translation that maps a type 1 structure M over Φ to a type 2 structure M^{12} over Φ^A such that*

1. *$M \models \varphi$ iff $M^{12} \models \varphi^{12}$;*
2. *if φ^{12} is satisfiable in a type 2 structure, then it is satisfiable in a special type 2 structure;*
3. *if φ^{12} is satisfiable in a special type 2 structure, then it is satisfiable in a type 1 structure.*

Thus, φ is valid iff φ^{12} is valid.

Proof: We first sketch the construction of φ^{12}. Assume without loss of generality that the sequences $\vec{x}$ that appear in subformulas of φ of the form $w_{\vec{x}}(\psi)$ are all distinct. If $\vec{x} = \langle x_{i_1}, \ldots, x_{i_n} \rangle$, we recursively replace subterms of the form $w_{\vec{x}}(\psi)$ with $w(\psi[x_{i_1}/a_{i_1}, \ldots, x_{in}/a_{in}])$. Let φ'' be the resulting statement. Suppose that $a_1, \ldots, a_N$ are the new constants that appear in φ''. Let `id` be the statement that says that the probabilities of the a_i's are identically distributed:

$$\mathtt{id} =_{\text{def}} \forall y \wedge_{j,k \in \{1,\ldots,N\}} (w(a_j = y) = w(a_k = y)).$$

Let ind be the statement that says that the a_j's are independent:

$$\texttt{ind} =_{\text{def}} \forall y_1, \ldots, y_N(w(\wedge_{1 \leq i \leq N}(a_i = y_i)) = \prod_{1 \leq i \leq N} w(a_i = y_i)).$$

Let $P_1, \ldots, P_k$ be the predicate symbols that appear in φ, and let $f_1, \ldots, f_m$ be the function symbols that appear in φ. The following formulas say that the predicate symbol P gets the same interpretation at all states:

$$\forall \vec{x}[(\neg P(\vec{x}) \wedge w(P(\vec{x})) = 0) \vee (P(\vec{x}) \wedge w(P(\vec{x})) = 1)].$$

A similar formula achieves the same effect for function symbols. Let fixed be the conjunction of these formulas for all the function and predicate symbols in φ, and let

$$\varphi^{12} =_{\text{def}} \texttt{id} \wedge \texttt{ind} \wedge \texttt{fixed} \wedge \varphi''.$$

Suppose $M = (D, \pi, \mu)$; we now construct M^{12}. The domain of M^{12} is also D. The set of states in M^{12} is D^N, that is, N-tuples of elements in D. State $(d_1, \ldots, d_N)$ occurs with probability $\mu^N(d_1, \ldots, d_N)$; in this state, the constant symbols $a_1, \ldots, a_N$ take the values $d_1, \ldots, d_N$, respectively. In all states, the other predicate and function symbols have the interpretation given by π. It follows from this definition that $M \models \varphi$ iff $M^{12} \models \varphi^{12}$.

As for the second part of the claim, the formulas fixed and ind guarantee that if a type 2 structure M satisfies φ^{12}, then M is special as far the symbols appearing in φ^{12} are concerned. We can thus easily construct a special type 2 structure M' that agrees with M on the symbols that appear in φ^{12}. Since the truth of φ^{12} depends only on the semantics of the symbols that appear in φ^{12}, it follows that φ^{12} is satisfied in M'. We leave details to the reader.

Finally, it is simple to turn a special type 2 structure into an "equivalent" type 1 structure: the type 1 structure has the same domain, the same interpretation of the symbols in Φ, and gives to d the probability of the set of worlds where a_1 takes the value d. Again, we leave details to the reader. ∎

Note that the type 2 structure M^{12} constructed in the proof depends on the formula φ (in particular, it depends on the number of variables that appear in the vector $\vec{x}$ in subformulas of φ of the form $w_{\vec{x}}\psi$). This dependence of M^{12} on φ results from our requirement that the probability over the possible worlds in M^{12} be discrete. We could have a uniform construction, but this would result in non-discrete probabilities over the continuum many worlds in D^ω. We remark that a similar construction (resulting in a non-discrete probability) was used by Gaifman [Gai64, Theorem 3].

We can also translate back from $\mathcal{L}_2$ to $\mathcal{L}_1$, with a simpler approach. The intuitive idea for getting the translation in this direction is fairly straightforward, and resembles usual constructions for modal logics. Given a set Φ of function and predicate symbols, let Φ^* be the result of replacing every predicate P (resp. function f) of arity n that appears in Φ with a predicate P^* (resp. function f^*) of arity $n+1$. Intuitively, the extra argument will range over possible worlds.

Theorem 6.2: *There is an effective translation that maps a formula φ in $\mathcal{L}_2(\Phi)$ (resp. $\mathcal{L}_2^=(\Phi)$) to a formula φ^{21} in $\mathcal{L}_1(\Phi^*)$ (resp. $\mathcal{L}_1^=(\Phi^*)$), and translations mapping a type 2 structure M over Φ to a type 1 structure M^{21} over Φ^* and from a type 1 structure N over Φ^* to a type 2 structure N' over Φ such that*

1. *$M \models \varphi$ iff $M^{21} \models \varphi^{21}$;*
2. *$N \models \varphi^{21}$ iff $N' \models \varphi$.*

Thus, φ is valid iff φ^{21} is valid.

Proof: Given a formula φ in $\mathcal{L}_2$, let φ^{21} be the result of replacing atomic formulas such as $P(x_1,\ldots,x_n)$ in φ by $P^*(x_1,\ldots,x_n,s)$, and replacing subterms of the form $w(P(x_1,\ldots,x_n))$ by $w_s(P^*(x_1,\ldots,x_n,s))$; we similarly replace a term such as $f(t_1,\ldots,t_n)$ that appears in ψ with the corresponding term $f^*(t_1^*,\ldots,t_n^*,s)$. Given a type 2 probability structure $M=(S,D,\pi,\mu)$, we consider a type 1 probability structure $M^{21}=(S\oplus D,\pi',\mu')$ with the following properties. The domain $S\oplus D$ is the disjoint union of S and D. We choose π' so that $(d_1,\ldots,d_n)\in\pi(s)(P)$ iff $(d_1,\ldots,d_n,s)\in\pi'(P^*)$; similarly $\pi(s)(f)(d_1,\ldots,d_n)=e$ iff $\pi'(f^*)(d_1,\ldots,d_n,s)=e$. We take μ' to be such that $\mu'(s)=\mu(s)$ (thus, $\mu'(d)=0$ for $d\in D$). Intuitively, we have made the states part of the domain, so we can replace taking the probability over the states where $P(x_1,\ldots,x_n)$ holds with taking the probability over the domain elements for which $P^*(x_1,\ldots,x_n,s)$ holds. It is now easy to check that $M\models\varphi$ iff $M^{21}\models\varphi^{21}$.

Given an arbitrary type 1 structure $N=(D,\pi,\mu)$ over Φ^*, let $N'=(D,D,\pi',\mu)$ be a type 2 structure over Φ, where π' is such that $(N',d)\models P(d_1,\ldots,d_n)$ iff $N\models P^*(d_1,\ldots,d_n,d)$ and $(N',d)\models f(d_1,\ldots,d_n)=e$ iff $N\models f(d_1,\ldots,d_n,d)=e$. We leave it to the reader to check that $N\models\varphi^{21}$ iff $N'\models\varphi$. ∎

7 Conclusions

We have investigated complexity and expressiveness issues for two related first-order logics of probability, where in one case we put the probability on the domain and, in the other, we put the probability on the states. We have shown that in general, the validity problem is highly intractable for both logics. All our results were proved under the assumption that the probability is discrete. It is easy to see that all our lower bounds go through (without change) if we allow arbitrary probability distributions. On the other hand, as we mentioned in the proof of Theorem 5.2, our upper-bound proofs for all the undecidability results do depend on the discreteness of the probability distribution. We conjecture that, in fact, the complexity of the logics gets even worse if we allow arbitrary probability distributions. Note that non-discrete probability distributions arise quite naturally in the context of type 2 structures (when considering an infinite sequence of coin tosses, for example).

One implication of these results is that we will not be able to find recursive axiom systems for these logics that are sound and complete (since the existence of such an axiom system would imply that the validity problem would be r.e.). There are a few special cases where our results show that it is possible to get complete axiomatizations, for example, in the case of $\mathcal{L}_1(\Phi)$ where Φ consists only of unary predicates and the case where we restrict to bounded domains. In a companion paper [Hal90], a sound set of axioms is provided for reasoning about probabilities over the domain and another is provided for reasoning about probabilities over possible worlds. These axioms are, in some sense, complete whenever possible. In particular, when combined with the standard axioms for reasoning about first-order logic, the axioms for reasoning about probabilities over the domain are complete for $\mathcal{L}_1(\Phi)$ if Φ contains only unary predicates; when combined with axioms for equality and an axiom that says that the domain has at most N elements, the axioms are complete for $\mathcal{L}_1(\Phi)$ if we restrict attention to domains with at most N elements.

It may very well be that for most applications we do not need the full power of first-order logic. Perhaps there are some interesting subclasses of the language for which validity is decidable. Unfortunately, our lower-bound proofs show that it does not take much to get a language which is badly undecidable. This is an issue that deserves further investigation.

Acknowledgements

A number of people helped improve both the content and presentation of this paper. We would like to thank Moshe Vardi for his many useful comments on the paper and for catching an error in an earlier draft of the paper, where we claimed a better upper bound for the complexity of the decidability problem. Haim Gaifman and Yishai Feldman read the paper very carefully and provided numerous suggestions for improvements. Danny Dolev made a valuable observation that helped to prove that $\mathcal{L}_2(\Phi)$ with even one unary predicate is Π_1^2 hard. Ron Fagin pointed out the Julia Robinson result [Rob49] used in the proof of Theorem 5.5. The second author would also like to thank Fahiem Bacchus for stimulating discussions on first-order logics of probability. Finally, Cynthia Hibbard and Mark Manasse gave us valuable editorial help.

References

[Bac88] F. Bacchus. On probability distributions over possible worlds. In *Proc. Fourth Workshop on Uncertainty in Artificial Intelligence*, pages 15–21, 1988.

[Bac90] F. Bacchus. *Representing and Reasoning with Probabilistic Knowledge*. MIT Press, Cambridge, Mass., 1990.

[BKR86] M. Ben-Or, D. Kozen, and J. H. Reif. The complexity of elementary algebra and geometry. *Journal of Computer and System Sciences*, 32(1):251–264, 1986.

[Car50] R. Carnap. *Logical Foundations of Probability*. University of Chicago Press, Chicago, 1950.

[CK90] C. C. Chang and H. J. Keisler. *Model Theory*. North-Holland, Amsterdam, 3rd edition, 1990.

[DG79] B. Dreben and W. D. Goldfarb. *The Decision Problem: Solvable Classes of Quantificational Formulas*. Addison-Wesley, Reading, Mass., 1979.

[End72] H. B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, New York, 1972.

[Fel84] Y. A. Feldman. *Probabilistic programming logics*. PhD thesis, Weizmann Institute of Science, 1984.

[Fen67] J. E. Fenstad. Representations of probabilities defined on first order languages. In J. N. Crossley, editor, *Sets, Models and Recursion Theory: Proceedings of the Summer School in Mathematical Logic and Tenth Logic Colloquium*, pages 156–172, 1967.

[FH84] Y. Feldman and D. Harel. A probabilistic dynamic logic. *Journal of Computer and System Sciences*, 28:193–215, 1984.

[FHM90] R. Fagin, J. Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Information and Computation*, 87(1/2):78–128, 1990.

[Gai60] H. Gaifman. Probability models and the completeness theorem. In *International Congress of Logic Methodology and Philosophy of*

Science, pages 77–78, 1960. This is the abstract of which [Gai64] is the full paper.

[Gai64] H. Gaifman. Concerning measures in first order calculi. *Israel Journal of Mathematics*, 2:1–18, 1964.

[GKP88] G. Georgakopoulos, D. Kavvadias, and C. H. Papadimitriou. Probabilistic satisfiability. *Journal of Complexity*, 4(1):1–11, 1988.

[GS74] S. Garfunkel and J. H. Schmerl. The undecidability of theories of groupoids with an extra predicate. *Proc. AMS*, 42(1):286–289, 1974.

[GS82] H. Gaifman and M. Snir. Probabilities over rich languages, testing and randomness. *Journal of Symbolic Logic*, 47(3):495–548, 1982.

[Hal90] J. Y. Halpern. An analysis of first-order logics of probability. *Artificial Intelligence*, 46:311–350, 1990.

[Hal91] J. Y. Halpern. Presburger arithmetic with unary predicates is Π^1_1 complete. *Journal of Symbolic Logic*, 56(2):637–642, 1991.

[Hin78] P. G. Hinman. *Recursion-Theoretic Hierarchies*. Springer-Verlag, Berlin/New York, 1978.

[Hoo78] D. N. Hoover. Probability logic. *Annals of Mathematical Logic*, 14:287–313, 1978.

[Kei85] H. J. Keisler. Probability quantifiers. In J. Barwise and S. Feferman, editors, *Model-Theoretic Logics*, pages 509–556. Springer-Verlag, Berlin/New York, 1985.

[Lew79] H. R. Lewis. *Unsolvable Classes of Quantificational Formulas*. Addison-Wesley, New York, 1979.

[Łoś63] J. Łoś. Remarks on the foundations of probability. In *Proc. 1962 International Congress of Mathematicians*, pages 225–229, 1963.

[Nil86] N. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.

[Rob49] J. Robinson. Definability and decision problems in arithmetic. *Journal of Symbolic Logic*, 14:162–186, 1949.

[Rog67] H. Rogers, Jr. *Theory of Recursive Functions and Effective Computability.* McGraw-Hill, New York, 1967.

[Tar55] A. Tarski. A lattice-theoretic fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–309, 1955.

[Tra50] B. A. Trakhtenbrot. Impossibility of an algorithm for the decision problem in finite classes. *Doklady Akademii Nauk SSSR*, 70:569–572, 1950.