Unifying Boolean and Algebraic Descriptive Complexity

Baptiste Chanus
Université Sorbonne Paris Nord, LIPN, CNRS, Villetaneuse, France

Damiano Mazza
CNRS, LIPN, Université Sorbonne Paris Nord, Villetaneuse, France

Morgan Rogers
Université Sorbonne Paris Nord, LIPN, CNRS, Villetaneuse, France

Abstract

We introduce ultrarings, which simultaneously generalize commutative rings and Boolean lextensive categories. As such, they allow to blend together standard algebraic notions (from commutative algebra) and logical notions (from categorical logic), providing a unifying descriptive framework in which complexity classes over arbitrary rings (as in the Blum, Schub, Smale model) and usual, Boolean complexity classes may be captured in a uniform way.

2012 ACM Subject Classification Theory of computation → Complexity theory and logic; Theory of computation → Complexity classes; Theory of computation → Finite Model Theory

Keywords and phrases Descriptive complexity theory, Categorical logic, Blum-Shub-Smale complexity

Digital Object Identifier 10.4230/LIPIcs.FSCD.2025.10

1 Introduction

Descriptive complexity measures the difficulty of computational problems in terms of the expressiveness that a logical language needs to describe them, rather than in terms of the power that a machine needs to solve them. Initiated in the 1970s with Fagin's theorem [14], which provides a characterization of the class NP, it has since been extended to capture virtually every other so-called "syntactic" complexity class [17].

Computational problems are usually defined as sets of finite objects (graphs, numbers, programs...) encoded as binary strings. Descriptive complexity regards such objects as totally ordered finite structures of a first-order language $\mathfrak{S}$. A formula φ over an extension $\mathfrak{T}$ of $\mathfrak{S}$ plays the role of a machine, accepting a finite structure s of $\mathfrak{S}$ if s is a model of φ. The more expressive $\mathfrak{T}$ is, the more powerful the corresponding machine. Notice that $\mathfrak{T}$ is typically *not* a first-order extension: for example, NP corresponds to allowing second-order existential quantifiers [14], and P corresponds to allowing fixpoint operators [16].

Computability and complexity may be meaningfully extended beyond finite objects. An interesting, well-known example is Blum, Shub and Smale's (BSS) model of computation over the real numbers [5]. Rather than having a tape whose cells contain either 0 or 1, BSS machines have a "tape" whose cells contain arbitrary real numbers. Usual Boolean operations are replaced by field operations, and testing the order relation $\leq$ plays the role of a conditional. Problems become sets of finite sequences of real numbers, and complexity may be developed in the usual way, including a definition of P, NP and complete problems.

Grädel and Meer [15] showed how the descriptive approach may be applied to the BSS model. The idea is to introduce a kind of hybrid finite structure, part logical and part algebraic. The algebraic part takes care of the data manipulated by the machine (real numbers and field operations), the logical part describes the discrete steps of the BSS

10th International Conference on Formal Structures for Computation and Deduction (FSCD 2025).
Editor: Maribel Fernández; Article No. 10; pp. 10:1–10:0
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

machine, allowing the transfer of Fagin's [14] and Immerman's [16] logical characterizations of P and NP to the BSS setting.

We provide a framework in which descriptive complexity may be uniformly formulated over Booleans and over real numbers (or other commutative rings) using a single logico-algebraic language. This language arises from a combination of categorical logic and commutative algebra, taking the form of *ultrarings*, a new algebraic object whose introduction is the main contribution of this paper.

To understand ultrarings, one may start from categorical logic. This is a vast subject [21, 18, 19], so we concentrate on what is relevant for our purposes. Define a *Boolean lextensive category* to be a category with finite limits and finite pullback-stable disjoint coproducts in which every subobject has a complement. A primordial example is **Fin**, the category of finite sets. Call a functor preserving finite limits and finite coproducts *logical.* Categorical logic gives us a correspondence between Boolean lextensive categories and *classical finite-limit theories* (or *cartesian theories* in Johnstone's terminology [19]). These are theories of first-order classical logic whose axioms are closed formulas of the form $\forall x_1 \dots \forall x_n . \varphi$ with φ quantifier-free.[1] The correspondence is as follows:

- from a classical finite-limit theory $\mathfrak{T}$ one can construct a small Boolean lextensive category $\mathrm{Syn}(\mathfrak{T})$, the *syntactic category* of $\mathfrak{T}$;
- each small Boolean lextensive category $\mathcal{B}$ induces a classical finite-limit theory $\mathrm{Lang}(\mathcal{B})$ such that $\mathrm{Syn}(\mathrm{Lang}(\mathcal{B}))$ is equivalent to $\mathcal{B}$;
- one may define *models* of $\mathfrak{T}$ in any Boolean lextensive category $\mathcal{K}$, and models of $\mathfrak{T}$ in $\mathcal{K}$ correspond to logical functors $\mathrm{Syn}(\mathfrak{T}) \to \mathcal{K}$.

In the last point, taking $\mathcal{K} = \mathbf{Fin}$ yields finite models in the usual sense, which suggests a categorical approach to descriptive complexity. In fact, there is a classical finite-limit theory $\mathfrak{Str}$ such that its finite models, modulo isomorphism, are exactly binary strings. Moreover, an extension $\mathfrak{T}$ of $\mathfrak{Str}$ induces a logical functor $\mathrm{Syn}(\mathfrak{Str}) \to \mathrm{Syn}(\mathfrak{T})$ (intuitively, an injection), so if we have a binary string seen as a functor $s : \mathrm{Syn}(\mathfrak{Str}) \to \mathbf{Fin}$, we may ask whether it factors through $\mathrm{Syn}(\mathfrak{T})$, which is to say whether s may be extended to a model of $\mathfrak{T}$. This means that extensions of $\mathfrak{Str}$ express computational problems (*i.e.*, subsets of $\{0,1\}^*$). The results of §4.2 capture the classes P and NP as certain forms of extensions.

The above framework only expresses Boolean computation. However, the fact that classical finite-limit theories may be regarded as presentations for their Boolean lextensive syntactic categories is analogous to the presentation of a ring by a set X equipped with a set S of integer polynomials with variables in X. The analogy is as follows:

- each presentation (X, S) induces a commutative ring $\mathbb{Z}[X]/\langle S\rangle$;
- each commutative ring R admits a presentation (X_R, S_R) such that $\mathbb{Z}[X_R]/\langle S_R\rangle \cong R$;
- given a commutative ring k, a *solution* in k of the system $\{p = 0\}_{p\in S}$ is the same thing as a ring homomorphism $\mathbb{Z}[X]/\langle S\rangle \to k$.

Notice how finding a model of a theory whose set of (closed) axioms is S may be seen as "solving the system" $\{\neg\varphi = 0\}_{\varphi\in S}$, where 0 is false, so the last point still fits the analogy.

Ultrarings turn the above analogy into two instances of a common construction. That is, ultrarings simultaneously generalize Boolean lextensive categories and commutative rings, and we have a notion of *annular theory* of which classical finite-limit theories and ring presentations are special cases. An annular theory is given by (possibly empty) sets of *sorts*, *generators* over the sorts, and *equations.* The latter are formulated using *polynomials*, which are a generalization of first-order formulas. Commutative rings are special "sortless" ultrarings: in

[1] Actually, φ may contain *provably unique* existentials, but this is irrelevant at the present level of detail.

case the are no sorts, a polynomial is exactly a polynomial in the usual algebraic sense. So ultrarings generalize commutative rings by adding sorts, and they generalize first-order logic by allowing ring operations on truth values.

Technically, an ultraring is a certain kind of small category (Definition 3), and the categories of commutative rings and of small Boolean lextensive categories are full subcategories of the category of ultrarings: to see a commutative ring R as an ultraring, we take the category of free R-modules of finite rank; to see a Boolean lextensive category $\mathcal{B}$ as an ultraring, we take the Kleisli category of the partiality monad over $\mathcal{B}$ (see §2.2).

In particular, there is an ultraring $\mathbb{R}$ corresponding to the field of real numbers and an ultraring $\mathbb{F}_1$ corresponding to the category of finite sets ($\mathbb{F}_1$ is in fact the initial ultraring). Let $\mathbb{K}$ be either $\mathbb{R}$ or $\mathbb{F}_1$. We will show that there is an annular theory $\mathfrak{Str}$ and an associated ultraring $\mathbb{K}[\mathfrak{Str}]$ such that, modulo isomorphism, a morphism $\mathbb{K}[\mathfrak{Str}] \to \mathbb{K}$ is a finite sequence of real numbers if $\mathbb{K} = \mathbb{R}$, or a binary string if $\mathbb{K} = \mathbb{F}_1$. The former is a typical input of a BSS machine over $\mathbb{R}$, the latter of a Turing machine. This gives a first idea of how both algebraic and Boolean data may be treated uniformly.

Computability and complexity may be expressed in ultrarings by means of *algebras*: given an ultraring $\mathcal{R}$, an $\mathcal{R}$-algebra is (roughly) an ultraring $\mathcal{A}$ together with a morphism $\mathcal{R} \to \mathcal{A}$. A $\mathbb{K}[\mathfrak{Str}]$-algebra $\mathcal{A}$ expresses a computational problem $\Gamma_{\mathbb{K}}(\mathcal{A})$ as follows: a string $\mathbb{K}[\mathfrak{Str}] \to \mathbb{K}$ is in $\Gamma_{\mathbb{K}}(\mathcal{A})$ iff it factors through $\mathcal{A}$. Our main result (Theorem 22) identifies classes of $\mathbb{K}[\mathfrak{Str}]$-algebras whose associated problems yield the standard classes RE (recursively-enumerable), NP and P defined either in terms of BSS machines over $\mathbb{R}$ or in terms of Turing machines, depending on how $\mathbb{K}$ is instantiated.

In logical terms, a $\mathbb{K}[\mathfrak{Str}]$-algebra $\mathcal{A}$ may be presented by an extension $\mathfrak{T}$ of the theory $\mathfrak{Str}$, and $x \in \Gamma_{\mathbb{K}}(\mathcal{A})$ is equivalent to the fact that x, seen as a model of $\mathfrak{Str}$ in $\mathbb{K}$, may be extended to a model of $\mathfrak{T}$. This highlights a shift with respect to descriptive complexity: rather than changing logical language for each complexity class, we fix one language (in the Boolean case, what categorical logicians would call “finite-limit logic”) and express complexity by means of logical extensions of $\mathfrak{Str}$ in this same language.

The rest of the paper introduces ultrarings (§2), how they may be presented by annular theories in the above sense (§3), and how they may be used to uniformly formulate complexity classes over both Booleans and real numbers (§4). We conclude with some perspectives (§5).

2 Ultrarings

2.1 Main Definitions

In what follows, we often abbreviate composition of arrows $f \circ g$ by fg. If $(\mathcal{C}, \otimes, I, \gamma)$ is a symmetric monoidal category, where γ is its symmetry, we denote by $\beta : I \otimes I \to I$ the canonical isomorphism. By “symmetric monoidal functor” we will always mean *strong* symmetric monoidal functor (*i.e.*, tensor and unit are preserved up to iso).

A *Frobenius sesquimonoid* in a symmetric monoidal category is an object A equipped with arrows $\delta : A \to A \otimes A$, $\varepsilon : A \to I$, $\delta^* : A \otimes A \to A$ such that (A, δ, ε) is a commutative comonoid, δ^* is commutative ($\delta^* \gamma_{A,A} = \delta^*$), associative ($\delta^*(\mathrm{id} \otimes \delta^*) = \delta^*(\delta^* \otimes \mathrm{id})$, left inverse to δ ($\delta^* \delta = \mathrm{id}_A$) and verifies the Frobenius law $(\mathrm{id}_A \otimes \delta^*)(\delta \otimes \mathrm{id}_A) = \delta\delta^* = (\delta^* \otimes \mathrm{id}_A)(\mathrm{id}_A \otimes \delta)$.

Initial objects will be denoted by 0. Recall that a (binary) coproduct diagram over an object A is a pair of arrows $\iota_1 : A_1 \to A$, $\iota_2 : A_2 \to A$, called *injections*, such that, for every $f_1 : A_1 \to B$, $f_2 : A_2 \to B$ there is a unique arrow $[f_1, f_2] : A \to B$ such that $[f_1, f_2]\iota_i = f_i$. In that case, we write $A = A_1 \oplus A_2$.

A *distributive monoidal category* is a symmetric monoidal category with finite coproducts such that, for all objects A, B, C, $A \otimes 0$ is initial and the arrow $\chi_{A,B,C} := [\mathrm{id}_A \otimes \iota_1, \mathrm{id}_A \otimes \iota_2] : (A \otimes B) \oplus (A \otimes C) \to A \otimes (B \oplus C)$ is an isomorphism.

Recall that an initial object which is also terminal is called a *zero object* and that any category with a zero object has, for any pair of objects A, B, a zero morphism $0_{A,B} : A \to B$ obtained by composing the terminal and initial arrows $A \to 0 \to B$. Also, for any pair of objects A_1, A_2, the category has *coinjections*[2] defined by:

$$\iota_1^* := [\mathrm{id}_{A_1}, 0_{A_2,A_1}] : A_1 \oplus A_2 \to A_1, \qquad \iota_2^* := [0_{A_1,A_2}, \mathrm{id}_{A_2}] : A_1 \oplus A_2 \to A_2.$$

It immediately follows that these verify

$$\iota_i^* \iota_j = \begin{cases} \mathrm{id}_{A_i} & \text{if } i = j \\ 0_{A_j,A_i} & \text{if } i \neq j. \end{cases} \tag{1}$$

▶ **Definition 1.** *A* δ^*-category *is a distributive monoidal category* $(\mathcal{C}, \otimes, I, \gamma)$ *with a zero object such that each object* A *is equipped with a Frobenius sesquimonoid structure* $\delta_A, \varepsilon_A, \delta_A^*$ *satisfying, for every objects* A, B, A_1, A_2,

$$\begin{aligned} \delta_{A\otimes B} &= (\mathrm{id}_A \otimes \gamma_{A,B} \otimes \mathrm{id}_B)(\delta_A \otimes \delta_B) \qquad & \varepsilon_{A\otimes B} &= \beta(\varepsilon_A \otimes \varepsilon_B) \qquad & \delta_I &= \beta^{-1}, \\ \delta^*_{A\otimes B} &= (\delta^*_A \otimes \delta^*_B)(\mathrm{id}_A \otimes \gamma_{A,B} \otimes \mathrm{id}_B) & \varepsilon_I &= \mathrm{id}_I & \delta^*_I &= \beta, \end{aligned}$$

$$\delta^*_{A_1\oplus A_2}(\iota_i \otimes \iota_j) = \begin{cases} \iota_i \delta^*_{A_i} & \textit{if } i = j \\ 0_{A_i\otimes A_j, A_1\oplus A_2} & \textit{if } i \neq j \end{cases} \qquad \delta_{A_1\oplus A_2}\iota_i = (\iota_i \otimes \iota_i)\delta_{A_i} \qquad \varepsilon_{A_1\oplus A_2}\iota_i = \varepsilon_{A_i},$$

and such that, for all $f, g : A \rightrightarrows B$, $\delta^*_B(f \otimes \mathrm{id}) = \delta^*_B(g \otimes \mathrm{id})$ *implies* $f = g$.

A morphism $F : \mathcal{C} \to \mathcal{D}$ *of* δ^**-categories is a finite-coproduct-preserving symmetric monoidal functor with structural isomorphisms* $\varphi_{A,B} : FA \otimes FB \to F(A \otimes B)$ *and* $\psi : I \to FI$ *such that, for each object* A *of* $\mathcal{C}$, $F\delta_A = \varphi_{A,A}\delta_{FA}$, $F\delta^*_A = \delta^*_{FA}\varphi^{-1}_{A,A}$ *and* $F\varepsilon_A = \psi\varepsilon_{FA}$.

In the sequel, subscripts such as those in $\mathrm{id}_A, \delta_A, \varepsilon_A \dots$ will be dropped when unambiguously retrievable from the context.

▶ **Lemma 2.** *In a* δ^**-category, coinjections are jointly monic.*

Proof. See §A.1. ◀

By the above lemma, for every arrows $f_1 : B \to A_1$, $f_2 : B \to A_2$ of a δ^*-category, there is at most one arrow $\langle f_1, f_2 \rangle : B \to A_1 \oplus A_2$, which we call *pairing*, such that $\iota_i^* \langle f_1, f_2 \rangle = f_i$. Some pairings automatically exist. For instance, by (1), $\iota_1 : A_1 \to A_1 \oplus A_2$ is the pairing $\langle \mathrm{id}_{A_1}, 0_{A_1,A_2} \rangle$ and dually for ι_2. The pairing $\langle 0_{B,A_1}, 0_{B,A_2} \rangle$ is simply $0_{B,A_1\oplus A_2}$. More subtly, the inverse χ^{-1}_{A,B_1,B_2} of the distributivity isomorphism is the pairing $\langle \mathrm{id}_A \otimes \iota_1^*, \mathrm{id}_A \otimes \iota_2^* \rangle$, which can be deduced using the fact that injections into a coproduct are jointly epic.

▶ **Definition 3.** *An* ultraring *is a small* δ^**-category such that:*

1. disjoint pairings: *for every* $f : C \to A$ *and* $g : C \to B$, *if* $(f \otimes g)\delta = 0$ *then their pairing* $\langle f, g \rangle$ *exists;*
2. complements*: for every* $p : A \to I$ *such that* $(p \otimes p)\delta = p$, *there exist a unique* $\overline{p} : A \to I$ *such that* $(p \otimes \overline{p})\delta = 0$ *and* $\nabla\langle p, \overline{p} \rangle = \varepsilon$, *where* $\nabla : I \oplus I \to I$ *is the codiagonal.*

A morphism of ultrarings *is just a morphism of* δ^**-categories. We denote by* **URing** *the strict* 2*-category whose objects are ultrarings, whose arrows are morphisms between them and whose* 2*-cells are monoidal natural transformations.*

[2] We refer to these as coinjections rather than projections to avoid the reader assuming that the coproducts are also products.

2.2 Examples of Ultrarings

Let R be a commutative ring. We denote by $\mathcal{U}R$ the category of free R-modules of finite rank, and linear maps between them. This may be equivalently described as the category whose objects are natural numbers and whose arrows $n \to m$ are $m \times n$ matrices (m rows, n columns) with coefficients in R, and where composition is matrix multiplication.

This category is distributive monoidal: the (strict) monoidal structure is given by multiplication of natural numbers for objects and by the standard tensor product of matrices (also known as Kronecker product) for morphisms. Coproduct is addition and 0 is easily seen to be a zero object. The Frobenius sesquimonoid structure is as follows: δ_n is the $n^2 \times n$ matrix whose i-th column has the element 1 at position $n(i-1)+i$ and is zero everywhere else; ε_n is the row matrix $(1\dots 1)$; and δ_n^* is the transpose of δ_n.

Under matrix addition, $\mathcal{U}R$ is actually an additive category, which means it has biproducts, so point (1) of Definition 3 is immediate. For what concerns point (2), any $p : n \to 1$ is a row matrix $(p_1 \dots p_n)$, and the condition $(p \otimes p)\delta = p$ amounts to $p_i^2 = p_i$ for all i. It is then a matter of elementary calculations to show that $\overline{p} := (1-p_1 \dots 1-p_n)$ is the unique morphism $n \to 1$ verifying $(p \otimes \overline{p})\delta = 0$ and $\nabla\langle p, \overline{p}\rangle = \varepsilon$. So $\mathcal{U}R$ is an ultraring.

A ring homomorphism $h : R \to S$ induces a morphism of ultrarings $\mathcal{U}h : \mathcal{U}R \to \mathcal{U}S$ by applying h pointwise to matrices. It is well known (see for instance [7]) that every symmetric monoidal, finite-coproduct-preserving functor $\mathcal{U}R \to \mathcal{U}S$ is monoidally isomorphic to a functor of the form $\mathcal{U}h$ for some $h : R \to S$. Therefore, if we consider the category of commutative rings **CRing** to be a 2-category in which every hom-category is discrete, this defines a 2-functor $\mathcal{U} : \mathbf{CRing} \to \mathbf{URing}$, and we have:

▶ **Proposition 4.** $\mathcal{U} : \mathbf{CRing} \hookrightarrow \mathbf{URing}$ *is fully faithful.*

Let $\mathbf{Fin}_*$ be the category of finite sets and partial functions. Observe that the cartesian product induces a symmetric monoidal structure on $\mathbf{Fin}_*$ (which is not a categorical product), whose unit is the singleton set $\{*\}$, and that $\mathbf{Fin}_*$ is obviously distributive (coproducts are disjoint unions) and has a zero object (the empty set). A Frobenius sesquimonoid structure may be defined by letting $\delta_A : A \to A \times A$ be the diagonal, $\varepsilon_A : A \to \{*\}$ be the unique total function and $\delta_A^* : A \times A \to A$ be the function which is defined on (a, a') exactly when $a = a'$.

We invite the reader to check that, if $f : C \to A$ and $g : C \to B$ are partial functions, the condition $(f \otimes g)\delta = 0$ amounts to the domains of f and g being disjoint. In that case, the pairing $\langle f, g\rangle : C \to A \oplus B$ is the evident partial function whose domain is the union of the domains of f and g. Furthermore, observe that any partial function $p : A \to \{*\}$ satisfies $(p \otimes p)\delta = p$, and corresponds to a subset $P \subseteq A$. The partial function $\overline{p} : A \to \{*\}$ corresponding to the complement of P is easily seen to be the unique one verifying the desired properties of point (2) of Definition 3.

The above shows that $\mathbf{Fin}_*$ has the structure of an ultraring, except that it is not small. This is easily fixed: let $\mathbb{F}_1$ be the category whose objects are natural numbers and whose arrows $n \to m$ are partial functions $\{1, \dots, n\} \to \{1, \dots, m\}$, or equivalently, $m \times n$ matrices with coefficients in $\{0, 1\}$ such that 1 appears at most once in each column. As a skeleton of $\mathbf{Fin}_*$, $\mathbb{F}_1$ is an ultraring. In fact, it is the initial ultraring: since $\mathbb{F}_1$ is generated under finite coproducts by the monoidal unit 1 and its identity arrow, any finite-coproduct-preserving functor $F : \mathbb{F}_1 \to \mathcal{R}$ with $\mathcal{R}$ an ultraring is determined by the image of 1; since morphisms of ultrarings are monoidal, $F1 \cong I$ and $\mathbf{URing}(\mathbb{F}_1, \mathcal{R})$ is equivalent to the terminal category.

Observe that $\mathbf{Fin}_*$ is the Kleisli category of the partiality monad $-+\{*\}$ over the category **Fin** of finite sets and *total* functions. This is a special case of a more general situation. A category is *lextensive* if it has finite limits and disjoint, pullback-stable finite coproducts.

A lextensive category is called *Boolean* if every subobject has a complement; see [8]. A primordial example is **Fin**. In fact, for what concerns finite limits and finite coproducts, the objects of Boolean lextensive categories may be thought of as sets and their arrows as total functions. In particular, if $\mathcal{B}$ is a Boolean lextensive category whose terminal object is 1, the endofunctor $-+1$ is a monad and the Kleisli category $\mathcal{B}_*$ may again be seen as a category of "sets and partial functions". The Kleisli categories of the partiality monad on small Boolean lextensive categories may be characterized as certain ultrarings, which we introduce below.

▶ **Definition 5.** *An ultraring is* Boolean *if every arrow f satisfies $\delta f = (f \otimes f)\delta$ and if for every $p : A \to I$ there exists a unique coproduct diagram $(\iota_p, \overline{\iota}_p)$ over A such that $p = \varepsilon\iota_p^*$, $\overline{\iota}_p = \iota_{\overline{p}}$ and, for every arrow f, $\iota_{\varepsilon f} = (\mathrm{id} \otimes \varepsilon)\iota_{\varepsilon\delta^*(f\otimes\mathrm{id})}$.*

In the following, we denote by **BoolURing** the full sub-2-category of **URing** on Boolean ultrarings, and by **BoolLext** the 2-category of small Boolean lextensive categories, finitely continuous functors which preserve finite coproducts, and natural transformations.

▶ **Proposition 6.** *The categories* **BoolURing** *and* **BoolLext** *are equivalent.*

Proof. We omit this proof due to space constraints. ◀

3 Annular Theories and Models

3.1 Presentations of Ultrarings

Having introduced the motivating examples of ultrarings, we now wish to provide *presentations* of these objects, which will subsequently enable us to perform essential constructions on them, as well as providing the basis for a syntax having a natural semantics in ultrarings.

▶ **Definition 7.** *A* signature $\mathfrak{S}$ *is a pair* $(\mathrm{Sort}(\mathfrak{S}), \mathrm{Gen}(\mathfrak{S}))$ *such that:*

- $\mathrm{Sort}(\mathfrak{S})$ *is a set of* sorts*. A* tensor *of arity n is a sequence of n sorts $A_1 \otimes \cdots \otimes A_n$. The unique tensor of arity 0 is denoted by I.*
- $\mathrm{Gen}(\mathfrak{S})$ *is a set of* generators*, each with a* type *$g : T \to \bigoplus_{j=1}^{k} U_j$ where T, U_j are tensors.*

We fix a countably infinite set of variables, ranged over by $x, y, z \ldots$ and we use $\vec{x}, \vec{y}, \ldots$ to denote finite sequences of variables. Given a signature $\mathfrak{S}$, the *monomials* over $\mathfrak{S}$ are defined as follows:

$$p, q ::= (x =_A y) \mid (g^j(\vec{x}) = \vec{y}) \mid 1 \mid \overline{p} \mid pq \mid \textstyle\int_x p,$$

where A ranges over $\mathrm{Sort}(\mathfrak{S})$, $g : T \to \bigoplus_{j=1}^{k} U_j$ ranges over $\mathrm{Gen}(\mathfrak{S})$, $1 \le j \le k$ and the length of $\vec{x}$ and $\vec{y}$ matches that of T and U_j, respectively. We write $g^j(\vec{x})$, $(g^j = \vec{y})$ and g^j rather than $(g^j(\vec{x}) =)$, $(g^j() = \vec{y})$ and $(g^j() =)$, respectively. We abbreviate $(x_1 =_{A_1} y_1) \cdots (x_n =_{A_n} y_n)$ to $(\vec{x} =_{\vec{A}} \vec{y})$. The notation $\int_x a$ is a binder: x is bound in a and is subject to the usual renaming conventions. If $\vec{x} = x_1, \ldots, x_n$, we write $\int_{\vec{x}} a$ for $\int_{x_1} \cdots \int_{x_n} a$. When $\vec{x}$ is empty, $\int_{\vec{x}} a$ is just a. We call a monomial *positive* if it contains no instances of the negation operator $\overline{p}$.

Monomials generalize logical formulas. Intuitively, in the Boolean case, the monomials 1, $\overline{p}$, pq and $\int_x p$ could be written $\top$, $\neg p$, $p \wedge q$, $\exists x.p$, and the monomial $(g^j(\vec{x}) = \vec{y})$ corresponds to a relation symbol $g^j(\vec{x}, \vec{y})$ satisfying that $g^j(\vec{x}, \vec{y}) \wedge g^j(\vec{x}, \vec{z})$ implies $\vec{y} = \vec{z}$, *i.e.*, a functional relation. The equality monomial $(x =_A y)$ is already a standard Boolean formula.

A *pre-polynomial* is a formal finite sum of monomials, with 0 representing the empty sum. We can extend the operations of multiplication and integration in the definition of monomials (the last two operations) to pre-polynomials by assuming distributivity and linearity, respectively.

$$\frac{i \in \{1,2\}}{x_i : A \vdash (x_1 =_A x_2) :: x_{3-i} : A} \qquad \frac{g : T \to \bigoplus_{j=1}^{k} U_j \quad 1 \leq j \leq k}{\vec{x} : T \vdash (g^j(\vec{x}) = \vec{y}) :: \vec{y} : U_j} \qquad \frac{}{\vdash 1 ::} \qquad \frac{\Gamma \vdash p :: \Delta}{\Gamma, x : A \vdash p :: \Delta}$$

$$\frac{\Gamma, x : A, y : A \vdash p :: \Delta}{\Gamma, x : A \vdash p[x/y] :: \Delta} \qquad \frac{\Gamma \vdash p :: \Delta, \vec{x} : T \quad \vec{x} : T, \Gamma' \vdash q :: \Delta'}{\Gamma, \Gamma' \vdash \int_{\vec{x}} pq :: \Delta, \Delta'} \qquad \frac{\Gamma \vdash p :: \Delta, x : A}{\Gamma, x : A \vdash p :: \Delta} \qquad \frac{\Gamma \vdash p ::}{\Gamma \vdash \overline{p} ::}$$

Figure 1 Monomials in context. We omit a rule allowing to permute type assignments in contexts.

$$\begin{aligned}
\textstyle\int_x \int_y p &\approx_\Gamma \textstyle\int_y \int_x p & \textstyle\int_y (x = y) &\approx_{\Gamma, x:A} 1 \\
(x =_A y)p &\approx_{\Gamma, x:A, y:A} (x =_A y)p[x/y] & (x =_A x) &\approx_{\Gamma, x:A} 1 \\
(x =_A y) &\approx_{\Gamma, x:A, y:A} (y =_A x) & p + \overline{p} &\approx_\Gamma 1 \\
\textstyle\int_x pq &\approx_\Gamma p \textstyle\int_x q \quad \text{if } x \text{ is not free in } p & &
\end{aligned}$$

Figure 2 Basic equations for congruences on pre-polynomials. $p[x/y]$ denotes usual substitution.

A *context* is a finite list of *type assignments* of the form $x : A$ where x is a variable and A a sort, such that no variable appears twice. If $\vec{x} = x_1, \ldots, x_n$ is repetition-free and $T = \bigotimes_{i=1}^{n} A_i$, we also write the context $x_1 : A_1, \ldots, x_n : A_n$ as $\vec{x} : T$. In particular, in $\vec{x} : I$ the sequence $\vec{x}$ is empty. If Γ and Δ are contexts, Γ, Δ denotes their concatenation (the variables appearing in each are assumed disjoint).

A *monomial-in-context* is an expression of the form $\Gamma \vdash p :: \Delta$, where p is a monomial and Γ and Δ are contexts, which is derivable by means of the inductive rules of Fig. 1. If $\Gamma \vdash p :: \Delta$ is derivable, we say that p is *well-typed*, that Γ is a *domain* for p and that the context Γ, Δ *matches* p. Note that, thanks to the penultimate rule of Fig. 1, any context matching p is also a domain for it. *Pre-polynomials-in-context* are defined similarly, adding a rule to derive $\Gamma \vdash \sum_{i \in I} p_i :: \Delta$ whenever $\Gamma \vdash p_i :: \Delta$ is derivable for all $i \in I$ (in particular, $\Gamma \vdash 0 :: \Delta$ is derivable for any Γ, Δ). From now on, we will only consider well-typed monomials and pre-polynomials.

A *congruence* is a context-indexed family of equivalence relations $\approx := (\approx_\Gamma)_\Gamma$ on pre-polynomials matching Γ, closed under associativity and commutativity of multiplication pq, neutrality of 1, every equation of Fig. 2 and such that:

- $p \approx_\Gamma q$ implies $\overline{p} \approx_\Gamma \overline{q}$, $pr \approx_\Gamma qr$ for all r matching Γ and $p \approx_{\Gamma'} q$ for every context Γ' including the assignments of Γ, possibly in a different order;
- $p \approx_{\Gamma, x:A} q$ implies $\int_x p \approx_\Gamma \int_x q$;
- $p \approx_\Gamma q$ iff $p + r \approx_\Gamma q + r$ for any r matching Γ (we call this *cancellativity*).

Given a set S of equations of the form $p \approx_\Gamma q$ where p, q are pre-polynomials matching Γ, the *congruence generated by* S is the smallest congruence containing the equations of S.

In the sequel, if $\approx$ is a congruence and we write $p \approx q$, we mean $p \approx_\Gamma q$ for every Γ matching both p, q.

▶ **Lemma 8.** *For any congruence $\approx$ and any pre-polynomial p, $p^2 \approx p$ iff $p\overline{p} \approx 0$. Moreover, either condition implies $\overline{p}^2 \approx \overline{p}$.*

Proof. See §A.1. ◀

Given a congruence $\approx$ over a signature $\mathfrak{S}$, we define $\approx$*-compatibility* to be the context-indexed family of smallest symmetric relations $\frown_\Gamma$ on monomials of $\mathfrak{S}$ closed under the rules

$$\frac{pq \approx_{\Gamma,\Delta} 0}{p \frown_\Gamma q} \qquad \frac{}{(g^j(\vec{x}) = \vec{y}) \frown_{\vec{x}:T,\Delta} (g^{j'}(\vec{x}) = \vec{y}')}\ {\scriptstyle j \neq j'}$$

$$\frac{p \frown_{\vec{x}:T} q}{\int_{\vec{x},\vec{y}} spr \frown_\Gamma \int_{\vec{x},\vec{z}} tqr}\ (*) \qquad \frac{p' \approx_{\Gamma,\Delta} p \quad p \frown_\Gamma q \quad q \approx_{\Gamma,\Delta} q'}{p' \frown_\Gamma q'}$$

Figure 3 Compatibility of pre-polynomials. The side condition $(*)$ is: $\Gamma, \vec{y} : U, \vec{z} : V$ (for some U, V) are domains for r, s, t, respectively, and $\vec{x}$ are the outputs of r.

of Fig. 3 such that $p \frown_\Gamma q$ implies that Γ is a domain for both p and q. We write $p \frown q$ to mean $p \frown_\Gamma q$ for every Γ which is a domain for both p and q.

A $\approx$-*polynomial* (or simply *polynomial* when this is unambiguous) is a pre-polynomial p such that there exists Γ such that $q \frown_\Gamma r$ for all monomials q, r of p and such that, whenever the expression $\overline{q}$ appears in p, we have $q^2 \approx q$.

▶ **Definition 9.** *An* annular theory $\mathfrak{T}$ *(which we abbreviate to* theory *in the remainder) is a triple* $(\mathrm{Sort}(\mathfrak{T}), \mathrm{Gen}(\mathfrak{T}), \mathrm{Eq}(\mathfrak{T}))$ *where* $(\mathrm{Sort}(\mathfrak{T}), \mathrm{Gen}(\mathfrak{T}))$ *is a signature and* $\mathrm{Eq}(\mathfrak{T})$ *is a well-founded set of* equations *of the form* $p \approx_\Gamma q$ *where* p, q *are pre-polynomials matching* Γ*, such that, calling such an equation* e*, we have that* p *and* q *are* $\approx$*-polynomials with* $\approx$ *the congruence generated by the equations of* $\mathrm{Eq}(\mathfrak{T})$ *strictly below* e *in the order. We denote by* $\approx^{\mathfrak{T}}$ *the congruence generated by* $\mathrm{Eq}(\mathfrak{T})$.

▶ **Definition 10.** *A* $\approx$-matrix *(or simply* matrix*) of* type $\bigoplus_{k\in K} \vec{x}^k : T_k \to \bigoplus_{j\in J} \vec{y}^j : U_j$ *consists of a* $(J \times K)$*-indexed family of* $\approx$*-polynomials-in-context* $\left(\vec{x}^k : T_k \vdash p_{jk} :: \vec{y}^j : U_j\right)_{(j,k)\in J\times K}$ *which we abbreviate to* (p_{jk})*, such that,*

$$\textit{for every } j \neq j' \in J,\ k \in K, \textit{ and monomials } r \textit{ in } p_{jk} \textit{ and } r' \textit{ in } p_{j'k},\ r \frown_{\vec{x}^k:T_k} r'. \qquad (\dagger)$$

We always identify $\approx$*-matrices whose entries are equal modulo* $\approx$.

▶ **Lemma 11.** *Let* (p_{jk}) *and* (q_{hj}) *be* $\approx$*-matrices of type* $\bigoplus_{k\in K} \vec{x}^k : T_k \to \bigoplus_{j\in J} \vec{y}^j : U_j$ *and* $\bigoplus_{j\in J} \vec{y}^j : U_j \to \bigoplus_{h\in H} \vec{z}^h : V_h$*, respectively. Then, the family of pre-polynomials-in-context indexed by* $H \times J$ *defined by*

$$(q \circ p)_{hk} := \left(\vec{x}^k : T_k \vdash \sum_{j\in J} \textstyle\int_{\vec{y}^j} q_{hj} p_{jk} :: \vec{z}^h : V_h\right)_{(h,k)\in H\times K}$$

is a $\approx$*-matrix of type* $\bigoplus_{k\in K} \vec{x}^k : T_k \to \bigoplus_{h\in H} \vec{z}^h : V_h$.

Proof. See §A.1. ◀

Let $\mathfrak{T}$ be a theory. We define a category $\mathbb{F}_1[\mathfrak{T}]$ as follows:

- the objects are formal finite coproducts $\bigoplus_{j\in J} \vec{x}^j : T_j$ of contexts of $\mathfrak{T}$, modulo injective renaming of variables.
- If $A := \bigoplus_{k\in K} \vec{x}^k : T_k$ and $B := \bigoplus_{j\in J} \vec{y}^j : U_j$, $\mathbb{F}_1[\mathfrak{T}](A, B)$ is the set of $\approx^{\mathfrak{T}}$-matrices of type $A \to B$.
- Composition is given by Lemma 11, and id_A is the diagonal matrix indexed by $K \times K$ which, at position (k, k), is equal to $(\vec{x}^k =_{T_k} \vec{z}^k)$, where we used the fact that, since objects are defined up to variable renaming, by renaming each x^k_h to z^k_h, we may write A as $\bigoplus_{k\in K} \vec{z}^k : T_k$.

As defined, $\mathbb{F}_1[\mathfrak{T}]$ is not small, because the finite sets indexing the objects are arbitrary. This may be circumvented by restricting to finite subsets of a fixed, countably infinite set of indices. We leave this implicit for notational convenience, but treat $\mathbb{F}_1[\mathfrak{T}]$ as a small category.

▶ **Proposition 12.** $\mathbb{F}_1[\mathfrak{T}]$ *is an ultraring.*

Proof. We spell out the required structures and properties, without entering into the details. Here "polynomial", "matrix", etc. mean $\approx^{\mathfrak{T}}$-polynomial, $\approx^{\mathfrak{T}}$-matrix, etc.

Let $A := \bigoplus_{j \in J} \vec{x}^j.T_j$ and $B := \bigoplus_{k \in K} \vec{y}^k.U_k$ be generic objects of $\mathbb{F}_1[\mathfrak{T}]$. The (strict) symmetric monoidal structure of $\mathbb{F}_1[\mathfrak{T}]$ is given by $A \otimes B := \bigoplus_{(j,k) \in J \times K} \vec{x}^j \vec{y}^k.T_j \otimes U_k$ on objects, with I as unit. On morphisms, tensor is matrix product, valid by a similar argument to the proof of Lemma 11, but without the integrals. The empty sum 0 is a zero object, the all-zero matrix of suitable size is the zero morphism between any two objects. The coproduct of A and B (where we may suppose $J \cap K = \emptyset$ and $\vec{x}^j, \vec{y}^k$ to be disjoint) is given by $\bigoplus_{l \in J \cup K} \vec{z}^l.V_l$, where $\vec{z}^l$ and V_l are equal to $\vec{x}^j$ and T_j (resp. $\vec{y}^k$ and U_k) if $l \in J$ (resp. $l \in K$). This obviously satisfies strict distributivity. The injections ι_1, ι_2 are the block column matrices $\begin{pmatrix} \mathrm{id} \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ \mathrm{id} \end{pmatrix}$, respectively, whereas the copairing of two matrices P, Q is the block row matrix $(P \; Q)$.

For what concerns the Frobenius sesquimonoid structure, δ_A is the matrix indexed over $J^2 \times J$ whose column for $j \in J$ has the element $(\vec{x}^j = \vec{y}^j)(\vec{x}^j = \vec{z}^j)$ at position (j, j) and is zero everywhere else, where we are renaming variables so that $A \otimes A = \bigoplus_{j,j' \in J} \vec{y}^j \vec{z}^{j'}.T_j \otimes T_{j'}$. ε_A is the row matrix which is equal to the monomial 1 everywhere. δ_A^* is the transpose of δ_A.

Thanks to the above definition, elementary calculations show that, if $P = (p_{kj})_{j \in J, k \in K}$ and $Q = (q_{lj})_{j \in J, l \in L}$ are matrices corresponding to morphisms $A \to B$ and $A \to C$ for some other object C (indexed by L, which we may supposed to be disjoint from K even if $C = B$), then $(P \otimes Q)\delta_A \approx^{\mathfrak{T}} 0$ is equivalent to asking that, for all $j \in J$, $k \in K$ and $l \in L$, we have $p_{kj} q_{lj} \approx^{\mathfrak{T}} 0$, which implies $p_{kj} \frown q_{lj}$, so we may form the block column matrix $R := \begin{pmatrix} P \\ Q \end{pmatrix}$ (with $K \cup L$ rows and J columns). The reader may check that the coinjections are the block row matrices (id 0) and (0 id), so R is the (unique) pairing of P, Q, as desired.

Let now $P = (p_k)_{k \in K}$ be a matrix of type $A \to I$. This means that P is a row matrix, and elementary calculations show that $(P \otimes P)\delta \approx^{\mathfrak{T}} P$ is equivalent to asking that $p_k^2 \approx^{\mathfrak{T}} p_k$ for all $k \in K$, so we are allowed to consider the polynomials $\overline{p_k}$ and define $\overline{P} := (\overline{p_k})_{k \in K}$. We need to show that $(P \otimes \overline{P})\delta \approx^{\mathfrak{T}} 0$. By similar calculations as above, this is equivalent to $p_k \overline{p_k} \approx^{\mathfrak{T}} 0$ for all $k \in K$, which holds by Lemma 8. We are left with proving that $\nabla \langle P, \overline{P} \rangle \approx^{\mathfrak{T}} \varepsilon$. Notice that, in general, $\nabla \langle Q, R \rangle = Q + R$ as matrices. This sum exists because, as shown above, $\langle Q, R \rangle$ is the block column matrix $\begin{pmatrix} Q \\ R \end{pmatrix}$. So we need to show that $p_k + \overline{p_k} \approx^{\mathfrak{T}} 1$ for all $k \in K$, but this is one of the equations of Fig. 2. ◀

3.2 Models and the Canonical Theory

The term 'annular theory' in Definition 9 is not merely an analogy with logic: it is chosen so that ultraring morphisms out of $\mathbb{F}_1[\mathfrak{T}]$ become *models* of $\mathfrak{T}$, in a sense that we now establish.

▶ **Definition 13.** *Given a signature $\mathfrak{S}$ and an ultraring $\mathcal{A}$, an $\mathfrak{S}$-*structure *in $\mathcal{A}$ consists of:*

- *An assignment $M_0 : \mathrm{Sort}(\mathfrak{S}) \to \mathrm{ob}(\mathcal{A})$. This extends to an assignment on tensors given by $M_0(\bigotimes_i A_i) := \bigotimes_i M_0(A_i)$.*
- *An assignment $M_1 : \mathrm{Gen}(\mathfrak{S}) \to \mathrm{mor}(\mathcal{A})$ such that if $g : T \to U$ then $M_1(g) : M_0(T) \to M_0(U)$.*

For $\mathfrak{S}$-structures M, M' in $\mathcal{A}$, a homomorphism $m : M \to M'$ *consists of a collection of morphisms $m_X : M_0(X) \to M'_0(X)$ in $\mathcal{A}$ indexed by $X \in \mathrm{Sort}(\mathfrak{S})$ such that for each $g : \bigotimes_i A_i \to \bigotimes_j B_j \in \mathrm{Gen}(\mathfrak{S})$, $M_1(g); \bigotimes_j m_{B_j} = \bigotimes_i m_{A_i}; M'_1(g)$.*

Given a theory $\mathfrak{T}$ over a signature $\mathfrak{S}$ and a $\mathfrak{S}$-structure M, we can define an interpretation over M of any $\approx^{\mathfrak{T}}$-polynomial-in-context, as detailed in §A.2. The notion of model is then defined in the expected way:

▶ **Definition 14.** *Let $\mathfrak{T}$ be a theory over signature $\mathfrak{S}$ and $\mathcal{A}$ an ultraring. A* model *of $\mathfrak{T}$ in $\mathcal{A}$ is a $\mathfrak{S}$-structure M in $\mathcal{A}$ such that for each equation $p \approx_{\Gamma,\Delta} q$ of $\mathfrak{T}$, we have $M(\Gamma \vdash p :: \Delta) = M(\Gamma \vdash q :: \Delta)$.*

For models M, M' of $\mathfrak{T}$ in $\mathcal{A}$, a homomorphism $M \to M'$ *is simply a homomorphism of $\mathfrak{S}$-structures. We denote by $\mathfrak{T}\text{-}\mathrm{Mod}(\mathcal{A})$ the category of models of $\mathfrak{T}$ in $\mathcal{A}$ and homomorphisms between them.*

▶ **Proposition 15.** *Let $\mathfrak{T}$ be a theory. There is an $\mathcal{A}$-natural equivalence of categories $\mathfrak{T}\text{-}\mathrm{Mod}(\mathcal{A}) \simeq \mathbf{URing}(\mathbb{F}_1[\mathfrak{T}], \mathcal{A})$.*

Proof. See §A.2. ◀

Proposition 15 implies in particular the existence of a *generic model* of any annular theory $\mathfrak{T}$, corresponding to the identity functor on $\mathbb{F}_1[\mathfrak{T}]$, of which a model in any ultraring is an image by naturality.

If Proposition 12 shows us that we can generate ultrarings from theories, it may not be a surprise that every ultraring arises in this way up to equivalence.

▶ **Definition 16.** *We define the* canonical theory *of an ultraring $\mathcal{R}$ to be the theory $\mathfrak{T}$ where:*

- $\mathrm{Sort}(\mathfrak{T}) = \mathrm{ob}(\mathcal{R})$*, where we denote the sort corresponding to A by $\ulcorner A \urcorner$;*
- $\mathrm{Gen}(\mathfrak{T})$ *has a generator $\ulcorner g \urcorner : \bigotimes_{j\in J} \ulcorner A_j \urcorner \to \bigoplus_{k\in K} \bigotimes_{l\in L_k} \ulcorner B_{k,l} \urcorner$ for each morphism $g : \bigotimes_{j\in J} A_j \to \bigoplus_{k\in K} \bigotimes_{l\in L_k} B_{k,l}$ in $\mathcal{R}$,*[3]
- $\mathrm{Eq}(\mathfrak{T})$ *consists of equations:*
  - $(\ulcorner \mathrm{id} \urcorner(\vec{x}) = \vec{y}) \approx (\vec{x} = \vec{y})$ *for* $\ulcorner \mathrm{id} \urcorner : \bigotimes_{j\in J} \ulcorner A_i \urcorner \to \bigotimes_{j\in J} \ulcorner A_i \urcorner$,
  - $\int_{\vec{y}} (\ulcorner f \urcorner(\vec{x}) = \vec{y})(\ulcorner g \urcorner(\vec{y}) = \vec{z}) \approx (\ulcorner gf \urcorner(\vec{x}) = \vec{z})$ *for composable f, g in $\mathcal{R}$ (with no coproduct decomposition of the codomains).*
  - $(\ulcorner \iota_j^* g \urcorner(\vec{x}) = \vec{y}) \approx (\ulcorner g \urcorner^j(\vec{x}) = \vec{y})$.
  - $(\ulcorner f \otimes g \urcorner(\vec{x}, \vec{u}) = \vec{y}, \vec{v}) \approx (\ulcorner f \urcorner(\vec{x}) = \vec{y})(\ulcorner g \urcorner(\vec{u}) = \vec{v})$.
  - $(\ulcorner 0 \urcorner(\vec{x}) = \vec{y}) \approx_{\vec{x}:T,\vec{y}:U,\Gamma} 0$*, for any typing of $\ulcorner 0 \urcorner$.*
  - $(\ulcorner \delta \urcorner(\vec{x}) = \vec{x_1}, \vec{x_2}) \approx (\vec{x} = \vec{x}_1)(\vec{x} = \vec{x}_2)$
  - $\ulcorner \varepsilon \urcorner(\vec{x}) \approx_{\vec{x}:T,\Gamma} 1$
  - $(\ulcorner \delta^* \urcorner(\vec{x_1}, \vec{x_2}) = \vec{x}) \approx (\vec{x}_1 = \vec{x})(\vec{x}_2 = \vec{x})$.

▶ **Theorem 17.** *Let $\mathcal{R}$ be an ultraring and $\mathfrak{T}$ its canonical theory, as defined in Definition 16. Then $\mathcal{R} \simeq \mathbb{F}_1[\mathfrak{T}]$.*

Proof. See §A.2. ◀

[3] This is a mild abuse of notation, since the name $\ulcorner g \urcorner$ can refer to multiple generators depending on how the domain and codomain are decomposed. When not specified, the type will be the one with the minimal decomposition.

3.3 Examples of Annular Theories

In what follows, we let $\mathbb{K}$ denote either $\mathbb{F}_1$ or the ultraring corresponding to a commutative ring R with no non-trivial idempotent, *i.e.*, in which $r^2 = r$ implies $r \in \{0,1\}$ for all $r \in R$. Examples include any field or, more generally, any integral domain such as $\mathbb{Z}$.

We define a *predicate* of a theory $\mathfrak{T}$ to be a $\approx^{\mathfrak{T}}$-polynomial $p : T \to I$ such that $p^2 \approx^{\mathfrak{T}}_{\vec{x}:T} p$. Observe that, if M is a model of $\mathfrak{T}$ in $\mathbb{K}$, then $M(p)$ must be a row vector of zeros and ones: the equation $p^2 \approx^{\mathfrak{T}}_{\vec{x}:T} p$ becomes $(M(p) \otimes M(p))\delta_T = M(p)$, and we already observed that this is equivalent to all entries of $M(p)$ being idempotent. Furthermore, $M(T) = I^{\oplus n}$ for some n (all objects of $\mathbb{K}$ are of this form). If we see $M(T)$ as the set $\{1,\dots,n\}$, $M(p)$ may be identified with the subset of $M(T)$ of all i such $M(p)_i = 1$, justifying the terminology.

Predicates are closed under Boolean logic: 0 and 1 are predicates; if p and q are predicates, then pq and $\overline{p}$ are predicates (Lemma 8). Also, by the second row of Fig. 2, $(x =_A y)$ is always a predicate: $(x =_A y)^2 \approx_{x:A,y:A} (x =_A y)(x =_A x) \approx_{x:A,y:A} (x = y)$. Moreover, models in $\mathbb{K}$ are consistent with Boolean logic: if M and p are as above, $M(0) = \emptyset$, $M(1) = M(T)$, $M(pq) = M(p) \cap M(q)$, $M(\overline{p}) = M(T) \setminus M(p)$ and $M(\vec{x} =_T \vec{y})$ is the diagonal of $M(T) \times M(T)$.

In the sequel, when we say that a generator P is a predicate over T, we mean that $P : T \to I$ and that the equation $P^2 \approx^{\mathfrak{T}}_{\vec{x}:T} P$ is added to the theory. By the above, a monomial φ consisting entirely of generators which are predicates is like a Boolean formula and, if $\vec{x} : T$ matches φ, we may impose that $\forall \vec{x}.\varphi$ holds by adding the equation $\varphi \approx_{\vec{x}:T} 1$.

For example, we define the theory $\mathfrak{Ord}$ with one sort N and one predicate $\leq$ over $N \otimes N$ with equations making $\leq$ is a total order. We write $(x \leq y)$ rather than $\leq(x,y)$. A model of $\mathfrak{Ord}$ in $\mathbb{K}$ is precisely a finite total order; modulo iso, we may assume it to be of the form $[n] := \{1 < \cdots < n\}$. The theory $\mathfrak{Str}$ is obtained from $\mathfrak{Ord}$ by adding a generator $X : N \to I$ with no further equations. A model s of $\mathfrak{Str}$ in $\mathbb{K}$ is a finite total order indexing a row vector $s(X)$ of zeros and ones if $\mathbb{K} = \mathbb{F}_1$, or of elements of R otherwise. In other words, s represents a binary string when $\mathbb{K} = \mathbb{F}_1$, or a string of elements of R otherwise. We denote the i-th bit/element by $s(X)_i$.

4 Computation

4.1 Computability over Ultrarings

Let $\mathbb{K}$ be as in §3.3. A $\mathbb{K}$-*string* is a binary string if $\mathbb{K} = \mathbb{F}_1$, or a string over elements of R otherwise. A $\mathbb{K}$-*problem* is a set of $\mathbb{K}$-strings.

We define $\mathbb{K}$-*operations* to be the following functions $R \times R \to R$, where R is $\{0,1\}$ if $\mathbb{K} = \mathbb{F}_1$ or the underlying ring of $\mathbb{K}$ otherwise: $\mathbb{F}_1$-operations are conjunction, the constant function 1 and the function negating the first bit and discarding the second; otherwise, $\mathbb{K}$-operations are constant functions, addition and multiplication of R and, in case R is a field, also division (the value of a division by zero is chosen arbitrarily). We also define $\mathrm{Test}_{\mathbb{K}} \subseteq R$ to be the set of non-negative elements when R is $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$, or $\{0\}$ otherwise.

▶ **Definition 18.** *A* $\mathbb{K}$-RAM*, where RAM stands for* random access machine*, is a (possibly empty) list* $I_0,\dots,I_{m-1}$ *of* instructions*,* $m \in \mathbb{N}$*, chosen among:* $\mathsf{comp}(\mathsf{op})$ *where* op *is a* $\mathbb{K}$*-operation (a* computation*);* zeror*,* zerow*,* incr *or* incw *(an* update*);* $\mathsf{branch}(l)$ *where* $0 \leq l \leq m+1$*; and* copy*.*

A configuration *of the machine is a tuple* (i,s,r,w) *where* $0 \leq i \leq m+1$*,* s *is a* $\mathbb{K}$*-string, whose* j*-th element is denoted by* s_j *and said to be the* content *of register* j*, and* $r, w \in \mathbb{N}$*. The initial configuration is* $(0,s,0,0)$*, where* s *is the input of the machine. On configuration* (i,s,r,w)*, the machine acts as follows. If* $0 \leq i \leq m-1$*, then the next configuration* c *is* *determined according to I_i (the string s is considered to be padded with infinitely many 0's on the right, so s_j is defined for all $j \in \mathbb{N}$):*

- $I_i = \mathsf{comp}(\mathrm{op})$*: c is $(i+1, s', r, w)$, where $s'_0 = \mathrm{op}(s_0, s_1)$, and $s'_j = s_j$ for all $j > 0$;*
- $I_i = \mathsf{branch}(l)$*: c is (l, s, r, w) if $s_0 \in \mathrm{Test}_{\mathbb{K}}$ or $(i+1, s, r, w)$ otherwise;*
- $I_i = \mathsf{copy}$*: c is $(i+1, s', r, w)$ where $s'_w = s_r$ and $s'_j = s_j$ for all $j \neq w$;*
- $I_i = \mathsf{zeror}$, zerow, incr *or* incw*: c is $(i+1, s, 0, w)$, $(i+1, s, r, 0)$, $(i+1, s, r+1, w)$ or $(i+1, s, r, w+1)$, respectively.*

Otherwise, the machine accepts if $i = m$ and rejects if $i = m+1$.

It is immediate to see that a $\mathbb{K}$-RAM with $\mathbb{K}$ one of $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$ is equivalent to a BSS machine [5] over those rings, modulo a polynomial slowdown (because computation steps of BSS machines are rational functions of arbitrary finite arity). On the other hand, an $\mathbb{F}_1$-RAM is obviously equivalent to a deterministic Turing machine on the alphabet $\{0, 1\}$.

The notion of acceptance/decision of a $\mathbb{K}$-problem by a $\mathbb{K}$-RAM is as usual. A polytime $\mathbb{K}$-RAM is a $\mathbb{K}$-RAM which always terminates in polynomially many steps in the size of the input. We denote by $\mathsf{RE}_{\mathbb{K}}$ the class of $\mathbb{K}$-problems accepted by a $\mathbb{K}$-RAM, by $\mathsf{P}_{\mathbb{K}}$ the class of $\mathbb{K}$-problems decided by a polytime $\mathbb{K}$-RAM, and by $\mathsf{NP}_{\mathbb{K}}$ the class of $\mathbb{K}$-problems which are projections over the first component of a $\mathbb{K}$-problem in $\mathsf{P}_{\mathbb{K}}$ whose $\mathbb{K}$-strings encode pairs (s_1, s_2) such that the length of s_2 is polynomially bounded by the length of s_1. By the above observations, $\mathsf{RE}_{\mathbb{F}_1}$, $\mathsf{P}_{\mathbb{F}_1}$ and $\mathsf{NP}_{\mathbb{F}_1}$ are the usual classes RE, P and NP, whereas $\mathsf{RE}_{\mathbb{R}}$, $\mathsf{P}_{\mathbb{R}}$ and $\mathsf{NP}_{\mathbb{R}}$ are the usual classes defined using the BSS model [5].

By contrast, $\mathsf{P}_{\mathbb{Z}}$ and $\mathsf{NP}_{\mathbb{Z}}$ do *not* coincide with the corresponding BSS classes. Indeed, BSS sort of "tweak" their definition of computability over $\mathbb{Z}$ in order to recover the usual, Boolean P and NP: they consider an integer to have size equal to its representation as a binary string. We do not need to do this, as we use $\mathbb{F}_1$ to capture Boolean computation.

4.2 Capturing Complexity Classes

Let $\mathcal{R}$ be an ultraring. We define an *$\mathcal{R}$-algebra* to be an ultraring $\mathcal{A}$ equipped with a *structure map* $\mathcal{R}' \to \mathcal{A}$ and an equivalence $\mathcal{R}' \simeq \mathcal{R}$. Typical examples are given by extensions of theories: if $\mathfrak{T}$ is defined by adding sorts and/or generators and/or equations to a theory $\mathfrak{S}$, then there is an obvious inclusion functor $\mathbb{F}_1[\mathfrak{S}] \to \mathbb{F}_1[\mathfrak{T}]$, mapping every object and morphism of $\mathbb{F}_1[\mathfrak{S}]$ to "itself", making $\mathbb{F}_1[\mathfrak{T}]$ into an $\mathbb{F}_1[\mathfrak{S}]$-algebra, with equality as equivalence. In general, if $\mathcal{R}$ is presented by $\mathfrak{S}$ with a chosen equivalence $e : \mathbb{F}_1[\mathfrak{S}] \simeq \mathcal{R}$, and if $\mathfrak{T}$ extends $\mathfrak{S}$, to make it clear that we are viewing $\mathbb{F}_1[\mathfrak{T}]$ as an $\mathcal{R}$-algebra with inclusion as structure map and equivalence e, we write $\mathcal{R}[\mathfrak{T} \setminus \mathfrak{S}]$ rather than $\mathbb{F}_1[\mathfrak{T}]$.

For example, if $\mathbb{K}$ is as in §3.3, we may fix a presentation $\mathfrak{T}_{\mathbb{K}}$ of $\mathbb{K}$ and an equivalence $\mathbb{F}_1[\mathfrak{T}_{\mathbb{K}}] \simeq \mathbb{K}$ (taking $\mathfrak{T}_{\mathbb{F}_1}$ to be the empty theory), and add to $\mathfrak{T}_{\mathbb{K}}$ the sort, generators, and equations of $\mathfrak{Str}$, obtaining a theory $\mathfrak{Str}_{\mathbb{K}}$ (so $\mathfrak{Str}_{\mathbb{F}_1} = \mathfrak{Str}$) and a $\mathbb{K}$-algebra that we denote by $\mathbb{K}[\mathfrak{Str}]$, which is equal to $\mathbb{F}_1[\mathfrak{Str}_{\mathbb{K}}]$ with structure map the inclusion of $\mathbb{F}_1[\mathfrak{T}_{\mathbb{K}}] \simeq \mathbb{K}$.

▶ **Definition 19.** *Let $\mathcal{A}$ be an $\mathbb{F}_1[\mathfrak{Str}]$-algebra with structure map $f : \mathcal{S} \to \mathcal{A}$ and equivalence $e : \mathcal{S} \to \mathbb{F}_1[\mathfrak{Str}]$. We say that $\mathcal{A}$* accepts *a $\mathbb{K}$-string s if any morphism $\mathbb{F}_1[\mathfrak{Str}] \to \mathbb{K}$ corresponding to s as a model of $\mathfrak{Str}$ in $\mathbb{K}$ (via Proposition 15) factors through f, modulo isomorphism. That is, given a morphism $g : \mathbb{F}_1[\mathfrak{Str}] \to \mathbb{K}$ representing s, there exists a morphism $h : \mathcal{A} \to \mathbb{K}$ and a monoidal natural isomorphism between $g \circ e$ and $h \circ f$. We denote by $\Gamma_{\mathbb{K}}(\mathcal{A})$ the set of $\mathbb{K}$-strings accepted by $\mathcal{A}$.*

Notice that the above definition does not depend on the choice of g representing s: if h exists for *some* g, then it exists for *all* g' representing s, because in that case $g' \cong g$ and therefore $g' \circ e \cong g \circ e \cong h \circ f$ as well.

Also observe that, by initiality of $\mathbb{F}_1$, every $\mathbb{K}[\mathfrak{Str}]$-algebra is an $\mathbb{F}_1[\mathfrak{Str}]$-algebra. Moreover, if $\mathbb{K}$ has no non-identity endomorphism (for instance, if $\mathbb{K}$ is $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ or $\mathbb{F}_p$ with p prime or 1), $\mathbb{K}$-strings still correspond to morphisms $\mathbb{K}[\mathfrak{Str}] \to \mathbb{K}$. In this case, the above definition may be equivalently formulated by speaking of $\mathbb{K}[\mathfrak{Str}]$-algebras. Concretely, if $\mathcal{A} = \mathbb{K}[\mathfrak{T}]$, a $\mathbb{K}$-string s, seen as a model of $\mathfrak{Str}$ in $\mathbb{K}$, is in $\Gamma_{\mathbb{K}}(\mathcal{A})$ iff we may extend it to a model in $\mathbb{K}$ of $\mathfrak{Str}$ augmented with the extra sorts, generators and equations of $\mathfrak{T}$. This is how the definition is applied in the statement of Theorem 22 below. The above definition is more general because, if $\mathbb{K}$ has non-trivial endomorphisms (for example $\mathbb{K} = \mathbb{C}$), a morphism $\mathbb{K}[\mathfrak{Str}] \to \mathbb{K}$ may not correspond to a $\mathbb{K}$-string, but a morphism $\mathbb{F}_1[\mathfrak{Str}] \to \mathbb{K}$ always does.

We say that an $\mathcal{R}$-algebra of the form $\mathcal{R}[\mathfrak{T}]$ is

- of *finite presentation* if $\mathfrak{T}$ is finite (finitely many sorts, generators and equations are added);
- *plain* if $\mathfrak{T}$ has no sorts (no sorts are added).

The latter terminology is justified by the fact that plain $\mathcal{R}$-algebras are algebras in the usual sense when $\mathcal{R}$ is a commutative ring.

Let T be a tensor of a theory $\mathfrak{T}$. We say that is it a *chain* if there exist a predicate (as defined in §3.3) $\leq_{\mathfrak{T}}$ over $T \otimes T$ and $\approx^{\mathfrak{T}}$-polynomials $\mathrm{Z}_T, \mathrm{Max}_T : I \to T$ and $\mathrm{Succ}_T : T \to T$ such that, with respect to $\approx^{\mathfrak{T}}$, $\leq_T$ is a total order, Z_T and Max_T are functions identifying the minimum and maximum element w.r.t. $\leq_{\mathfrak{T}}$, and Succ_T is the successor function (every element has a unique successor and predecessor, except zero and max, which have no predecessor and successor, respectively). Notice that, if T and U are chains, the so is $T \otimes U$ via the lexicographic order (it is not hard to see that this is definable in annular theories). Whenever p is a polynomial matching a context $\Gamma, \vec{t} : T$, we will write $p(0)$, $p(\max)$ and $p(\vec{t}_{+1})$ to mean $\fint_{\vec{t}} \mathrm{Z}_T(\vec{t})p(\vec{t})$, $\fint_{\vec{t}} \mathrm{Max}_T(\vec{t})p(\vec{t})$ and $\fint_{\vec{t}'} \mathrm{Succ}_T(\vec{t}, \vec{t}')p(\vec{t}')$, respectively.

▶ **Definition 20.** *Let $\mathcal{R}$ be an ultraring and $\mathcal{A} = \mathcal{R}[\mathfrak{T}]$ a plain $\mathcal{R}$-algebra, whose underlying equivalence is $\mathbb{F}_1[\mathfrak{S}] \simeq \mathcal{R}$. We say that $\mathcal{A}$ is* inductive *if there exists a tensor T of $\mathfrak{S}$ which is a chain in $\mathfrak{T}$ such that every added generator $g \in \mathrm{Gen}(\mathfrak{T})$ is either making T into a chain or has type $T \otimes U \to V$ for some U and V and belongs to one of the following classes:*

- inductive generators*: generators with equations of the form*
  - initialization*: $\sum_i \varphi_i g(0) \approx_\Gamma \sum_i \varphi_i c_i$,*
  - induction*: $\sum_i \psi_i(\vec{t}) g(\vec{t}_{+1}) \approx_{\Gamma, \vec{t}:T} \sum_i \psi_i(\vec{t}) f_i(\vec{t})$,*

  and, possibly, equations stating that g is a predicate and/or final value *equations of the form $\sum_i \varphi'_i g(\max) \approx_\Gamma \sum_i \varphi'_i c'_i$, where c_i, c'_i, f_i are polynomials and $\varphi_i, \varphi'_i, \psi_i$ are predicates such that, for all $i \neq j$, $\varphi_i \varphi_j \approx^{\mathfrak{T}}_\Gamma 0$, $\varphi'_i \varphi'_j \approx^{\mathfrak{T}}_\Gamma 0$ and $\psi_i \psi_j \approx^{\mathfrak{T}}_{\Gamma, \vec{t}:T} 0$;*
- harmless predicates *and* ancillary generators*: predicates P over T coming with their* ancillary generators *$X^P_1, \ldots, X^P_k : T \to I$ and their* defining equations *$Pp \approx_{\vec{t}:T} 0$ and $\overline{P} q_i \approx_{\vec{t}:T} 0$ for $1 \leq i \leq k$, where p and q_i are polynomials such that:*
  - *the degree of p in the ancillary generators of P is at most 2;*
  - *no harmless predicate appears in p or any q_i;*
  - *in any model M of $\mathfrak{S}$ in $\mathbb{K}$, the equation $M(p) = 0$ has a solution in $\mathbb{K}$ (with respect to the ancillary generators) iff the system $(M(q_i) = 0)_{i=1}^k$ has no solution in $\mathbb{K}$.*

  Moreover, the ancillary generators of P do not appear in any equation of $\mathfrak{T}$ except the defining equations of P.

Inductive algebras are inspired by fixed point operators in logic [16, 15]. Let $\mathbb{K}$ be an integral domain or $\mathbb{F}_1$. The idea is that, given a model M in $\mathbb{K}$ of the base theory $\mathfrak{S}$ (*i.e.*, a morphism $f : \mathcal{R} \to \mathbb{K}$), finding whether M may be extended to a model M' of $\mathfrak{S}$ plus the generators and equations of $\mathfrak{T}$ (*i.e.*, whether f factors via the inclusion $\mathcal{R} \to \mathcal{A}$) may be done inductively on T, which is like a time parameter. Indeed, to build M' one must find an interpretation in $\mathbb{K}$ for the generators of $\mathfrak{T}$. For an inductive generator, the initialization equation gives the value when $\vec{t} = 0$ and, once the value at $\vec{t}$ is known, the induction equation gives the value for $\vec{t}_{+1}$. For a harmless predicate P with defining equations involving p and (q_i), one may know whether P holds at time $\vec{t}$ by testing whether $p = 0$ has a solution in $\mathbb{K}$. This may be done efficiently (at least for certain $\mathbb{K}$ of interest, such as $\mathbb{R}$ and $\mathbb{F}_1$) because p has degree at most two (it involves something like computing a discriminant, or solving a 2-CNF). The definition guarantees that, if $p = 0$ has a solution, then one of the $q_i = 0$ does not, therefore $\overline{P}(\vec{t}) = 0$ (because $\mathbb{K}$ has no zero divisor) hence P is true at time $\vec{t}$. Otherwise, if $p = 0$ has no solution, then P must be false at time $\vec{t}$.

The following results hold more generally than stated (for example, they also hold for $\mathbb{Q}$, $\mathbb{Z}$ and $\mathbb{F}_p$ with p prime), but for succinctness we prove them only for $\mathbb{F}_1$ and $\mathbb{R}$.

▶ **Lemma 21.** *Let $\mathbb{K}$ be $\mathbb{F}_1$ or $\mathbb{R}$. For any polytime $\mathbb{K}$-RAM M there exists a finite extension $\mathfrak{X}$ of $\mathfrak{Str}$ such that:*

- *$\mathbb{K}[\mathfrak{X}]$ is an inductive $\mathbb{K}[\mathfrak{Str}]$-algebra of finite presentation;*
- *a model of $\mathfrak{X}$ in $\mathbb{K}$ is an accepting run of M, and its restriction to $\mathfrak{Str}$ is the input of M for that run.*

Proof. Let the running time of M be bounded by n^k, with n the input length and k a constant. The theory $\mathfrak{X}$ adds a chain structure on N (the sort of $\mathfrak{Str}$), so that $T := N^k$ is also a chain, which will play the role of the time parameter. Then, $\mathfrak{X}$ adds the inductive generators $\mathrm{Reg} : T \otimes N^k \to I$ (the work tape, which is of length at most n^k because it cannot be longer than the running time of M), $\mathrm{P}_r, \mathrm{P}_w : T \to N^k$ (the values of r and w), and the inductive predicates $\mathrm{In}_0, \dots, \mathrm{In}_m$ over T (where $\mathrm{In}_i(t)$ holds iff instruction i is executed at time t). If $\mathbb{K} = \mathbb{R}$, $\mathfrak{X}$ also adds the harmless predicate Gez over T, with ancillary generators $\mathrm{Pos}, \mathrm{Neg}, \mathrm{Neg}^{-1} : T \to I$ (for the $\mathrm{Test}_{\mathbb{K}}$ relation), and the harmless predicate Div over T, with ancillary generator $\mathrm{Quo} : T \to I$ (for division).

The equations added by $\mathfrak{X}$ are as follows (we will explain them momentarily):

$$\mathrm{In}_0(0) \approx 1, \quad \mathrm{In}_k(0) \approx 0 \text{ for } k \neq 0 \quad \max(t) \approx \mathrm{In}_m(t) \quad \mathrm{In}_i(t)\mathrm{In}_j(t) \approx 0 \text{ for all } i \neq j \tag{2}$$

$$\mathrm{Gez}(t)(\mathrm{Pos}(t)^2 - \mathrm{Reg}(t,0)) \approx 0 \quad \overline{\mathrm{Gez}(t)}(\mathrm{Neg}(t)^2 + \mathrm{Reg}(t,0)) \approx 0 \quad \overline{\mathrm{Gez}(t)}(\mathrm{Neg}(t)\mathrm{Neg}^{-1}(t) - 1) \approx 0 \tag{3}$$

$$\mathrm{Div}(t)(\mathrm{Reg}(t,0) - \mathrm{Reg}(t,1)\mathrm{Quo}(t)) \approx 0 \quad \overline{\mathrm{Div}(t)}\mathrm{Reg}(t,0) \approx 0 \tag{4}$$

$$(\mathrm{P}_w(0) = 0) \approx 1 \quad (\mathrm{P}_r(0) = 0) \approx 1 \tag{5}$$

$$\int_x \mathrm{In}_k(t)(\mathrm{P}_i(t) = x)\overline{(\mathrm{P}_i(t_{+1}) = x_{+1}) + (\mathrm{P}_i(t_{+1}) = 0)} \approx 0 \text{ with } i \in \{r, w\} \tag{6}$$

$$\int_x \mathrm{In}_k(t)(\mathrm{P}_i(t) = x)\overline{(\mathrm{P}_i(t_{+1}) = x)} \approx 0 \tag{7}$$

$$\mathrm{In}_0(t)\overline{\mathrm{In}_1(t_{+1})} + \cdots + \mathrm{In}_k(t)\mathrm{Gez}(t)\overline{\mathrm{In}_l(t_{+1})} + \mathrm{In}_k(t)\overline{\mathrm{Gez}(t)\mathrm{In}_{k+1}(t_{+1})} + \cdots + \mathrm{In}_{m-1}(t)\overline{\mathrm{In}_m(t_{+1})} \approx 0 \tag{8}$$

$$\mathrm{In}_k(t)\mathrm{Reg}(t_{+1},0) \approx \mathrm{In}_k(t)(\mathrm{Reg}(t,0) * \mathrm{Reg}(t,1)) \text{ where } * \in \{+,-,\times\} \tag{9}$$

$$\mathrm{In}_k(t)\mathrm{Reg}(t_{+1},0) \approx \mathrm{In}_k(t)\mathrm{Div}(t)(\mathrm{Reg}(t,1)/\mathrm{Reg}(t,0)) + \mathrm{In}_k(t)\overline{\mathrm{Div}(t)} \tag{10}$$

$$\int_x \mathrm{In}_k(t)(\mathrm{P}_w(t) = x)\mathrm{Reg}(t_{+1},x) \approx \int_y \mathrm{In}_k(t)(\mathrm{P}_r(t) = y)\mathrm{Reg}(t,y) \tag{11}$$

$$\overline{(\textstyle\sum_{i\in I} \mathrm{In}_i(t)) + (\sum_{j\in C} \mathrm{In}_j(t))(\mathrm{P}_w(t) = 0)}\mathrm{Reg}(t,0) \approx \overline{(\textstyle\sum_{i\in I} \mathrm{In}_i(t)) + (\sum_{j\in C} \mathrm{In}_j(t))(\mathrm{P}_w(t) = 0)}\mathrm{Reg}(t_{+1},0) \tag{12}$$

$$\int_s (\textstyle\sum_{i\in C} \mathrm{In}_i(t))(\mathrm{P}_w(t) \neq s)(s \neq 0)\mathrm{Reg}(t,s) \approx \int_s (\sum_{i\in C} \mathrm{In}_i(t))(\mathrm{P}_w(t) \neq s)(s \neq 0)\mathrm{Reg}(t_{+1},s) \tag{13}$$

$$(i <_T \max\nolimits_{\mathrm{N}})\mathrm{Reg}(0,i) + \overline{(i <_T \max\nolimits_{\mathrm{N}})}\mathrm{Reg}(0,i) \approx (i <_T \max\nolimits_{\mathrm{N}})X(i) \tag{14}$$

Equations (2) state that the first (resp. last) instruction is the instruction of index 0 (resp. m) and that only one instruction is executed at a time; (3) define the predicates Gez (which is true iff $s_0 \geq 0$, where s_0 is the content of register 0, as in Definition 18), Pos (resp. Neg and Neg^{-1}) which is a witness of positivity (resp. negativity); (4) define the predicates $\mathrm{Div}, \mathrm{Quo}$ which are witnesses of divisibility and the quotient; (5) and (6) state that P_w and P_r are initialized to 0 and how they are updated, note that, due to the chain structure of N^k, $x_{+1} \neq 0$ and thus $(\mathrm{P}_i(t) = x_{+1})(\mathrm{P}_i(t) = 0) \approx 0$ so the sum is well defined; (7) states that when In_k is not an update(i) the P_i addresses is not modified; (8) defines the jump instructions, (9) and (10) the operations and (11) the copy instructions; (12) states that the first cell of the tape is only modified by computations and copies, where I is the set of indices of computation and C the one of copy instructions, also, note that the sets I and C are disjoints which is important for summability; (13) states that the other cells can only change if the instruction is a copy and the writing register is the right one; (14) states that the work tape initially contains the input and is filled with zeros after that, note that $\max_{\mathrm{N}}$ is the maximum of the sort N (which $\mathfrak{X}$ equips with a chain structure), whereas $<_T$ is the strict order on the tensor $T = N^k$ (which is lexicographically ordered from the order on N).

By definition, $\mathfrak{X}$ adds no sorts and is finite. Towards proving the second point of the statement, we observe that a finite model of $\mathfrak{X}$ consists of a $\mathbb{K}$-string x of length $|N|$ (the underlying model of $\mathfrak{Str}$ in $\mathbb{K}$), where $|N|$ is the interpretation of N, plus seven additional functions and $m + 3$ predicates.

Let us prove the second point of the statement by induction on $|N|^k$: if this is 1 (*i.e.*, there is only one time step), then by equations 1 and 2 we have that $m = 0$, so the machine has only one state, which is accepting. Therefore, any model is an accepting run of the machine. If $|N|^k = \{t_0, \cdots, t_{n+1}\}$ (note that, by definition, $(t_n)_{+1} = t_{n+1}$), by induction all the functions and predicates are defined at time t_n, then, for some i we have that $\mathrm{In}_i(t_n)$ is defined. Also, i cannot be m by equation 2. This means that the current instruction is either a branch, an update, a copy or a computation.

First, if it is a branch(l), then, since we supposed that $\mathrm{Gez}(t_n)$ is defined, by equation 9 we have that either $\mathrm{In}_l(t_{n+1})$ or $\mathrm{In}_{i+1}(t_{n+1})$ is defined, in any of those cases, $\mathrm{In}_j(t_{n+1})$ is also defined by equation 3. By equations 13 and 14 we have that $\mathrm{Reg}(t_{n+1}, s)$ is also defined for any s. Axioms 4 and 5 ensure that $\mathrm{Gez}, \mathrm{Pos}, \mathrm{Neg}, \mathrm{Neg}^{-1}, \mathrm{Div}$ and Quo are defined at time t_{n+1}, and equation 8 gives us that $\mathrm{P}_r(t_{n+1})$ and $\mathrm{P}_w(t_{n+1})$ are both defined.

Then, if the instruction is an update, again, equations 13 and 14 ensure the definition of $\mathrm{Reg}(t_{n+1}, s)$ for any s, equations 4 and 5 for the definitions of $\mathrm{Gez}, \mathrm{Pos}, \mathrm{Neg}, \mathrm{Neg}^{-1}, \mathrm{Div}$ and Quo, equation 7 for $\mathrm{P}_j(t_{n+1})$ and 8 for $\mathrm{P}_{j'}(t_{n+1})$, with $j, j' \in \{r, w\}$ and $j \neq j'$.

If the instruction is a copy, since $\mathrm{P}_r(t_n)$ and $\mathrm{P}_w(t_n)$ are defined we have that, for some $s \in N^k$, $\mathrm{Reg}(t_{n+1}, s)$ is defined by equation 12, if $s \neq 0$ then $\mathrm{Reg}(t_{n+1}, 0)$ is defined thanks to equation 13 and for all other $y \in N^k$ $\mathrm{Reg}(t_{n+1}, y)$ is defined by equation 14. Again all other functions and predicates are defined by the same equations as before.

Finally, if the instruction is a computation, then either it is a division and $\mathrm{Reg}(t_{n+1}, 0)$ is defined by equation 11 ($\mathrm{Reg}(t, 0)$ is divisible by $\mathrm{Reg}(t, 1)$, if not then the behavior of the machine is not defined) either it is not a division and $\mathrm{Reg}(t_{n+1}, 0)$ is defined by equation 10. For all $s \neq 0, \mathrm{Reg}(t_{n+1}, s)$ is defined by equation 14 and the other functions and predicates are still defined as before.

We have thus proved that, at any given $t \leq |N|^k$, if i is the unique element of $\{0, \ldots, m\}$ such that $\mathrm{In}_i(t)$ holds, $(i, \mathrm{Reg}(t), \mathrm{P}_r(t), \mathrm{P}_w(t))$ is the configuration obtained by running M for t steps from the initial configuration on input x. Equation 2 guarantees that this is an accepting run, as the machine must have entered instruction m at some point before time $|N|^k$ (after which it stays there).

We are left with checking that the resulting algebra is inductive. The inductive generators are clearly of the right form, so we just need to check that the other predicates are indeed harmless: for Gez, in any model, at any time t either $\mathrm{Reg}(t,0)$ is positive and there exists a real number $\mathrm{Pos}(t)$ such that $\mathrm{Pos}(t)^2 - \mathrm{Reg}(t,0) = 0$ either it is negative and there exists $\mathrm{Neg}(t)$ and $\mathrm{Neg}^{-1}(t)$ such that $\mathrm{Neg}(t)\mathrm{Neg}^{-1}(t) = 1$ and $\mathrm{Neg}^2 + \mathrm{Reg}(t,0) = 0$, in any case $(\mathrm{Pos}(t)^2 - \mathrm{Reg}(t,0))(\mathrm{Neg}^2 + \mathrm{Reg}(t,0))$ and $(\mathrm{Pos}(t)^2 - \mathrm{Reg}(t,0))(\mathrm{Neg}\mathrm{Neg}^{-1} - 1)$ have a solution. The same is true for Div, either $\mathrm{Reg}(t,0)$ is 0, either is not and there is a real number $\mathrm{Quo}(t)$ such that $(\mathrm{Reg}(t,0) - \mathrm{Reg}(t,1)\mathrm{Quo}(t)) = 0$, in any case $\mathrm{Reg}(t,0)(\mathrm{Reg}(t,0) - \mathrm{Reg}(t,1)\mathrm{Quo}(t))$ has a solution. Hence, the theory is inductive. ◀

▶ **Theorem 22.** *Let $\mathbb{K}$ be $\mathbb{F}_1$ or $\mathbb{R}$ and let L be a $\mathbb{K}$-problem. Then:*

1. $L \in \mathsf{RE}_{\mathbb{K}}$ *iff* $L = \Gamma_{\mathbb{K}}(\mathcal{A})$ *with* $\mathcal{A}$ *a* $\mathbb{K}[\mathfrak{Str}]$*-algebra of finite presentation;*
2. $L \in \mathsf{NP}_{\mathbb{K}}$ *iff* $L = \Gamma_{\mathbb{K}}(\mathcal{A})$ *with* $\mathcal{A}$ *a plain* $\mathbb{K}[\mathfrak{Str}]$*-algebra of finite presentation;*
3. $L \in \mathsf{P}_{\mathbb{K}}$ *iff* $L = \Gamma_{\mathbb{K}}(\mathcal{A})$ *with* $\mathcal{A}$ *an inductive* $\mathbb{K}[\mathfrak{Str}]$*-algebra of finite presentation.*

Proof. For (1), consider an arbitrary $\mathbb{K}$-RAM M. We slightly modify the proof of Lemma 21: rather than using N^k as chain, we *introduce* a new sort T, with the structure of a chain, and reproduce the proof with T instead of N^k. The theory thus obtained is no longer inductive but is still of finite presentation, and its models are accepting runs of M, with the interpretation of T giving its running time, which may now be arbitrary.

Conversely, if we are given an algebra $\mathbb{K}[\mathfrak{T}]$ with $\mathfrak{T}$ finitely extending $\mathfrak{Str}$ and a $\mathbb{K}$-string s, we have that a model of $\mathfrak{T}$ extending s has finitely more data than s, namely the data interpreting the additional sorts and generators of $\mathfrak{T}$. Given such extra data, checking whether it verifies the additional equations of $\mathfrak{T}$, of which there are finitely many, is certainly doable in a finite time (in fact, in polynomial time) on a $\mathbb{K}$-RAM.

For (2), let $L \in \mathsf{NP}_{\mathbb{K}}$. This means that there exists a polytime $\mathbb{K}$-RAM M and a constant c such that, for every $\mathbb{K}$-string s, $s \in L$ iff there exists a string s' of length at most $|s|^c$ such that M accepts the pair (s, s'). We let k be the maximum between c and the exponent of the polynomial bound on the running time of M, and construct a $\mathbb{K}[\mathfrak{Str}]$-algebra just like in the proof of Lemma 21, expect that we do not initialize the work tape beyond the input length. The algebra thus obtained is still plain (no sort is added) of finite presentation, but it is no longer inductive because the generator Reg now is missing some initialization equations. The models of the resulting theory are still accepting runs of M, on which the tape is now initialized non-deterministically beyond the input s. In particular, it may be initialized with the witness s', which is small enough to fit in N^k.

The converse is much like point (1), except that this time we only need to account for the interpretation of the extra generators, which are of polynomial size (because these have types of the form $N^k \to \bigoplus_j N^{k_j}$ with k and k_j constants) and may therefore be guessed and then checked with a polytime $\mathbb{K}$-RAM.

For (3), the implication from left to right is Lemma 21. For the converse, if we have an inductive algebra $\mathbb{K}[\mathfrak{T}]$ with $\mathfrak{T}$ finitely extending $\mathfrak{Str}$, and we have a $\mathbb{K}$-string s, we check whether s may be extended to a model of $\mathfrak{T}$ as outlined after Definition 20. The only thing left to verify is that computing harmless generators takes no more than polynomial time.

Let us start with the case $\mathbb{K} = \mathbb{R}$. For each harmless predicate P we have an equation of the form $P(t)\varphi(t) \approx_\Gamma 0$, so for each t, to know whether $P(t)$ is true in a given model in $\mathbb{R}$ we just need to check whether the interpretation of $\varphi(t)$ has a root in $\mathbb{R}$. The idea here is to generalize the discriminant to multivariate polynomials of degree two. See the appendix for details (Lemma 24).

When $\mathbb{K} = \mathbb{F}_1$, the truth of a harmless generator at time t is controlled by a polynomial $\varphi(t)$ of degree two. In $\mathbb{F}_1$, this is interpreted as a 2-DNF. Therefore, testing the equation $\varphi(t) = 0$ is testing whether a 2-DNF is falsifiable, which is well-known to be polytime [20]. ◀

Let us conclude with a few words about when $\mathbb{K} = \mathbb{Z}$. In this case, Lagrange's four square theorem may be used to implement $\mathrm{Test}_{\mathbb{Z}}$. Moreover, combined with the Davis-Putnam-Robinson-Matiyasevich theorem, Theorem 22 implies that $\mathsf{RE}_{\mathbb{Z}} = \mathsf{NP}_{\mathbb{Z}}$.

5 Discussion and Perspectives

As implicitly shown in §2.2, ultrarings generalize commutative rings by generalizing their categories of modules (free of finite rank). This is not a new idea: it has already been mentioned or directly used in at least a few attempts to generalize algebraic geometry beyond commutative rings [13, 10, 6, 4]. What is new in ultrarings is the emphasis on presenting such generalized algebraic objects by means of a language which is directly derived from first-order logic and which still captures all ring presentations.

Speaking of algebraic geometry, it is very interesting to point out that the ultraring $\mathbb{F}_1$ has some of the properties typically expected of the "field with one element". In fact, the full subcategory of ultrarings which are plain $\mathbb{F}_1$-algebras (*i.e.*, of the form $\mathbb{F}_1[\mathfrak{T}]$ with $\mathfrak{T}$ sortless) may be fully and faithfully embedded in the category of generalized commutative rings studied by Durov [12]. Under this embedding, $\mathbb{F}_1$ is mapped to Durov's version of the "field with one element".

The reader may have observed that the field $\mathbb{F}_2$ could also have been used capture Boolean computation: $\mathbb{F}_2$-strings are just binary strings and $\mathsf{RE}_{\mathbb{F}_2}$, $\mathsf{NP}_{\mathbb{F}_2}$ and $\mathsf{P}_{\mathbb{F}_2}$ coincide with the usual Boolean classes. We contend that this is sort of an accident, and that Boolean computability is really over $\mathbb{F}_1$. Indeed, any $\mathbb{K}$-RAM may be restricted to operate only on $\{0, 1\}$, simulating a Boolean machine. This is routinely used in BSS or algebraic complexity to define the "Boolean part" of the language decided by a machine. Our approach nicely explains this by the existence of a unique morphism $\mathbb{F}_1 \to \mathbb{K}$ saying that "Booleans are everywhere". By contrast, there is no morphism $\mathbb{F}_2 \to \mathbb{K}$ unless $\mathbb{K}$ has characteristic 2.

In this paper, we have not developed the theory of ultrarings to the extent necessary to give a technical content to the above statement, but to the acquainted reader we may say that the above observation becomes a *change of base* along $\mathbb{F}_1 \to \mathbb{K}$. In fact, the definition of ultraring in this paper has been intentionally restricted to give "logical-looking" presentations. We already know that ultrarings may be given a more liberal definition, still allowing presentations, but of a different flavor than the ones introduced here. Modules for these ultrarings may open interesting perspectives in terms of computational complexity: at least in the plain case (*i.e.*, for the class NP and below), modules generalize algebras and could potentially be used to give finer characterizations of complexity classes.

In a different direction, if we restrict to Boolean lextensive categories (which, as §2.2 shows, are special cases of ultrarings) we know that more Boolean complexity classes may be captured by adapting ideas from descriptive complexity. These include deterministic and nondeterministic logarithmic space, the logtime hierarchy and the polynomial hierarchy, as well as non-uniform complexity classes, and will be the subject of another paper [9].

Finally, let us mention that descriptive complexity has well-known tools, taking the form of various pebble games [17], allowing one to establish lower bound results. These have been reformulated categorically [1, 3, 11, 2, 22]. Knowing if and how these reformulations interface with our work is an interesting question for the future.

References

1 Samson Abramsky, Anuj Dawar, and Pengming Wang. The pebbling comonad in finite model theory. In *Proceedings of LICS*, pages 1–12. IEEE Computer Society, 2017.

2 Samson Abramsky and Luca Reggio. Arboreal categories: An axiomatic theory of resources. *Log. Methods Comput. Sci.*, 19(3), 2023.

3 Samson Abramsky and Nihil Shah. Relating structure and power: Comonadic semantics for computational resources. *J. Log. Comput.*, 31(6):1390–1428, 2021.

4 John C. Baez, Joe Moeller, and Todd Trimble. 2-rig extensions and the splitting principle. arXiv:2410.05598 [math.CT], 2024.

5 Lenore Blum, Mike Shub, and Steve Smale. On a theory of computation and complexity over the real numbers. *Bulletin of the American Mathematical Society*, 21(1):1–46, 1989.

6 Martin Brandenburg. *Tensor categorical foundations of algebraic geometry.* Ph.d. thesis, University of Münster, 2014.

7 Martin Brandenburg and Alexandru Chirvasitu. Tensor functors between categories of quasi-coherent sheaves. *Journal of Algebra*, 399:675–692, 2014.

8 Aurelio Carboni, Stephen Lack, and R.F.C. Walters. Introduction to extensive and distributive categories. *Journal of Pure and Applied Algebra*, 84(2):145–198, 1993.

9 Baptiste Chanus and Damiano Mazza. A categorical approach to describing complexity classes. In preparation. Available on the authors' web page, 2024.

10 Alexandru Chirvasitu and Theo Johnson-Freyd. The fundamental pro-groupoid of an affine 2-scheme. *Appl. Categ. Structures*, 21(5):469–522, 2013.

11 Anuj Dawar, Tomás Jakl, and Luca Reggio. Lovász-type theorems and game comonads. In *Proceedings of LICS*, pages 1–13. IEEE, 2021.

12 Nikolai Durov. New Approach to Arakelov Geometry. arXiv 0704.2030 [math.AG], 2007.

13 Nikolai Durov. Classifying vectoids and operad kinds. *Proc. Steklov Inst. Math.*, 273:48–63, 2011.

14 Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In *Complexity of Computation, SIAM-AMS Proccedings*, volume 7, pages 43–73, 1974.

15 Erich Grädel and Klaus Meer. Descriptive complexity theory over the real numbers. In *Proceedings of STOC*, pages 315–324. ACM Press, 1995.

16 Neil Immerman. Relational queries computable in polynomial time. *Information and Control*, 68(1):86–104, 1986.

17 Neil Immerman. *Descriptive complexity.* Graduate texts in computer science. Springer, 1999.

18 Bart P. F. Jacobs. *Categorical Logic and Type Theory*, volume 141 of *Studies in logic and the foundations of mathematics.* North-Holland, 2001.

19 Peter T. Johnstone. *Sketches of en Elephant. A Topos Theory Compendium. Volume 2.* Oxford University Press, 2002.

20 Melven R. Krom. The decision problem for a class of first-order formulas in which all disjunctions are binary. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 13(1–2):15–20, 1967.

21 Saunders Mac Lane and Ieke Moerdijk. *Sheaves in Geometry and Logic.* Springer, 1994.

22 Yoàv Montacute and Glynn Winskel. Concurrent games over relational structures: The origin of game comonads. In *Proceedings of LICS*, pages 58:1–58:14. ACM, 2024.

A Appendix

A.1 Proofs of Some Lemmas

Proof of Lemma 2. We start by observing that, if $\iota_i : B_i \to B_1 \oplus B_2$ are injections, then $\iota_i \delta^*_{B_i}(\iota_i^* \otimes \mathrm{id}_{B_i}) = \delta^*_{B_1 \oplus B_2}(\mathrm{id}_{B_1 \oplus B_2} \otimes \iota_i)$. We prove this for $i = 1$, the other case is essentially identical. We omit obvious subscripts for identity arrows. Let $\chi = [\iota_1 \otimes \mathrm{id}, \iota_2 \otimes \mathrm{id}] : (B_1 \otimes B_1) \oplus (B_2 \otimes B_1) \to (B_1 \oplus B_2) \otimes B_1$ be the distributivity isomorphism. We have, using (1), $\iota_1 \delta^*_{B_1}(\iota_1^* \otimes \mathrm{id})\chi = \iota_1 \delta^*_{B_1}[\iota_1^*\iota_1 \otimes \mathrm{id}, \iota_1^*\iota_2 \otimes \mathrm{id}] = \iota_1 \delta^*_{B_1}[\mathrm{id}, 0] = [\iota_1 \delta^*_{B_1}, 0] = [\delta^*_{B_1 \oplus B_2}(\iota_1 \otimes \iota_1), \delta^*_{B_1 \oplus B_2}(\iota_2 \otimes \iota_1)] = \delta^*_{B_1 \oplus B_2}(\mathrm{id} \otimes \iota_1)\chi$, which proves the claim because χ is invertible. We will also need that, if $\kappa_i : A \otimes B_i \to (A \otimes B_1) \oplus (A \otimes B_2)$ are injections and $\chi' : (A \otimes B_1) \oplus (A \otimes B_2) \to A \otimes (B_1 \oplus B_2)$ another instance of distributivity, then $\chi'\kappa_i = \mathrm{id} \otimes \iota_i$, which holds by definition of χ'.

Suppose now that we have $f, g : A \rightrightarrows B_1 \oplus B_2$ with $\iota_i^* f = \iota_i^* g$ for $i = 1, 2$. Using the above observations, we have $\delta^*_{B_1 \oplus B_2}(f \otimes \mathrm{id})\chi'\kappa_i = \delta^*_{B_1 \oplus B_2}(f \otimes \iota_i) = \iota_i \delta^*_{B_i}(\iota_i^* f \otimes \mathrm{id}) = \iota_i \delta^*_{B_i}(\iota_i^* g \otimes \mathrm{id})$, from which, applying the first two equalities in reverse order, we obtain $\delta^*_{B_1 \oplus B_2}(f \otimes \mathrm{id})\chi'\kappa_i = \delta^*_{B_1 \oplus B_2}(g \otimes \mathrm{id})\chi'\kappa_i$ for $i = 1, 2$. Since injections are jointly epic and χ' is invertible, we infer $\delta^*_{B_1 \oplus B_2}(f \otimes \mathrm{id}) = \delta^*_{B_1 \oplus B_2}(g \otimes \mathrm{id})$, from which we conclude $f = g$ by definition δ^*-category. ◀

Proof of Lemma 8. We start by observing that cancellativity implies $\overline{\overline{p}} \approx p$, as well as $\overline{1} \approx 0$. Next, we claim that, whenever q matches the same contexts as p and $q^2 \approx q$, we have $\overline{pq} \approx \overline{p}q + p\overline{q} + \overline{p}\,\overline{q}$. Indeed, $pq + \overline{pq} \approx 1 \approx (p + \overline{p})(q + \overline{q}) \approx pq + \overline{p}q + p\overline{q} + \overline{p}\,\overline{q}$, so we conclude by cancellativity.

Suppose that $p\overline{p} \approx 0$. Then, $p \approx p(p + \overline{p}) \approx p^2 + p\overline{p} \approx p^2$. For the converse, if $p^2 \approx p$, then using the above claim we have $\overline{p\overline{p}} = p^2 + \overline{p}^2 + \overline{p}p = p + \overline{p}(\overline{p} + p) = p + \overline{p} = 1$, so $p\overline{p} = 0$ by the first two observations. Finally, we have $\overline{p} \approx \overline{p^2} \approx p\overline{p} + \overline{p}p + \overline{p}^2 \approx \overline{p}^2$. ◀

Proof of Lemma 11. For indices h, j, j', k, let r, r', s, s' be monomials in $p_jk, p_{j'k}, q_{hj}, q_{hj'}$, respectively. We need to check that each $(q \circ p)_{hk}$ is a polynomial by verifying that, when $j \neq j'$ we have $\int_{\vec{y}^j} sr \frown_{\vec{x}^k : T_k} \int_{\vec{y}^{j'}} s'r'$. By hypothesis, $r \frown_{\vec{x}^k : T_k} r'$, so we conclude by the last rule of Fig. 3. We must also check that the context is valid. By hypothesis, $q_{hj}p_{jk}$ is matched by $\vec{x}^k : T_k, \vec{y}^j : U_j, \vec{z}^h : V_h$, with $\vec{y}^j$ and $\vec{z}^h$ being outputs, hence each monomial $\int_{\vec{y}^j} rs$ is well-typed and matched by $\vec{x}^k : T_k, \vec{z}^h : V_h$, as required. Finally, we must check condition (†) of Definition 10. Given $h \neq h'$, we require $\int_{\vec{y}^j} sr \frown_{\vec{x}^k : T_k} \int_{\vec{y}^{j'}} t'r'$, where t' is a monomial of $q_{h'j'}$. If $j \neq j'$, we conclude as above. If $j = j'$, by hypothesis we have $s \frown_{\vec{y}^j : U_j} t'$, so we conclude by applying the last rule of Fig. 3 to these instead. ◀

A.2 Interpreting Polynomials in a $\mathfrak{S}$-structure

Let M be a $\mathfrak{S}$-structure. We start by inductively defining an interpretation over M for *positive* monomials-in-context, following Fig. 1.

- As a base case, we interpret monomials over $\mathfrak{S}$ with the maximal codomain context of all outputs and the minimal domain of all remaining free variables, for each of the basic constructions of monomials:
  - $M\,(x : A \vdash (x =_A y) :: y : A)$ is $\mathrm{id}_{M_0(A)}$,
  - $M\,\big(\vec{x} : T \vdash (g^j(\vec{x}) = \vec{y}) :: \vec{y} : U\big)$ is $M_1(g); \iota_j^*$,
  - $M\,(\vdash 1 ::)$ is $\varepsilon_I = \mathrm{id}_I$,
- Next, we describe the effect of expanding and reorganizing contexts.
  - Given $f := M(\Gamma \vdash p :: \Delta, x : A)$, let $M(\Gamma, x : A \vdash p :: \Delta) := \varepsilon_{M_0(A)} \delta^*_{M_0(A)}(f \otimes \mathrm{id}_A)$.

  - Relatedly, given $f := M(\Gamma \vdash p :: \Delta)$, we define $M(x : A, \Gamma \vdash p :: \Delta)$ to be $\beta(\varepsilon_{M_0(A)} \otimes f)$, (recalling that $\beta : I \otimes I \to I$ is the canonical isomorphism).
  - Given $M(\Gamma \vdash p :: \Delta)$, permutations of Γ and Δ are interpreted by pre- or post-composing with the corresponding symmetry morphisms.
- Finally, we specify how monomials combine.
  - Given $f := M(\Gamma \vdash p :: \Delta, \vec{x} : T)$ and $g := M(\vec{x} : T, \Gamma' \vdash q :: \Delta')$ we can define $M\left(\Gamma, \Gamma' \vdash \textstyle\int_{\vec{x}:T} pq :: \Delta, \Delta'\right)$ to be $(\mathrm{id} \otimes g)(f \otimes \mathrm{id})$; to recover an interpretation of $\int_{\vec{x}} p$, we identify this with $\int_{\vec{x}} p1$.
  - Given $f := M(x_1 : A, x_2 : A, \Gamma \vdash p :: \Delta)$ and a variable x not appearing in p, we define $M(x : A, \Gamma \vdash p[x_1/x][x_2/x] :: \Delta)$ to be $f(\delta_{M_0(A)} \otimes \mathrm{id})$.
  - Dually, given $f := M(\Gamma \vdash p :: x_1 : B, x_2 : B, \Delta)$ and a variable x not appearing in p, we define $M(\Gamma \vdash p[y_1/y][y_2/y] :: y : B, \Delta)$ to be $(\delta^*_{M_0(B)} \otimes \mathrm{id})f$.

Let us check that the above interpretations are well-defined. The first set of rules give unambiguous interpretations. For the second set, there is exactly one way to arrive a given context using these rules up to reordering, so it suffices to check that the order of operations does not affect the result; this is straightforward, relying on the fact that δ^* is symmetric.

The only place where ambiguity is introduced in the third set of rules, where the interpretation ostensibly depends on how the domain and codomain contexts are partitioned. We take these case by case.

- Consider $M\left(\Gamma, \Gamma' \vdash \int_{\vec{x}:T} pq :: \Delta, \Delta'\right)$. By disjointness, Δ and Δ' must respectively contain only outputs of p and q, so this is unambiguous; the remaining free variables of p and q must respectively go into Γ and Γ'. Any remaining free variables, whether assigned to Γ or Γ', must be interpreted by ε, which eliminates any remaining ambiguity.
- In $M(x : A, \Gamma \vdash p[x_1/x][x_2/x] :: \Delta)$, there is formally a choice to make about how to assign the instances of x to x_1 or x_2; however, since the inductive construction will eventually disambiguate between any pair of repeated instances, commutativity and associativity of δ guarantee that any such choice will result in the same interpretation (considering also that the interpretation is invariant under α-equivalence).
- Similarly, distinguishing duplicates of variables in the codomain produces a well-defined result by commutativity and associativity of δ^*, so we are done.

Next, given $f := M(\Gamma \vdash p ::)$, we have that $M\left(\Gamma \vdash p^2 ::\right)$ is exactly $(f \otimes f)\delta$. As such, when these interpretations are equal, we may take $M(\Gamma \vdash \overline{p} ::)$ to be $\overline{f}$ provided by Definition 3, thus extending the above to monomials featuring $\overline{p}$.

Finally, we define $M(\Gamma \vdash 0 :: \Delta)$ to be the zero morphism of the appropriate type.

▶ **Lemma 23.** *Suppose we are given a theory $\mathfrak{T}$ over a signature $\mathfrak{S}$ and a $\mathfrak{S}$-structure M in $\mathcal{A}$. Suppose that whenever we have positive monomials-in-context $\Gamma \vdash p :: \Delta$ and $\Gamma \vdash q :: \Delta$ such that $p \approx_{\Gamma,\Delta} q$, it is the case that $M(\Gamma \vdash p :: \Delta) = M(\Gamma \vdash q :: \Delta)$, so that in particular M can interpret all monomials which are $\approx$-polynomials. Suppose further that M identifies p and q whenever $p \approx q$ when p and q are either monomials or 0. Then $p \frown_{\mathfrak{T}} q$ implies the pairing $\langle p, q\rangle$ exists in $\mathcal{A}$.*

Proof. It suffices to check the axioms of Fig. 3 which determine the compatibility relation. Indeed, $pq \approx 0$ guarantees that if M respectively interprets p, q as f, g in a given context then the first axiom of ultrarings Definition 3 ensures that the pairing exists.

The pairing of two components j, j' of (the interpretation of) a generator g is the result of coinjecting g onto the coproduct of their codomains.

If $p \frown q$ and $\langle p, q\rangle$ exists, we can obtain the pairing of the interpretations of $\int_{\vec{x},\vec{y}} spr$ and $\int_{\vec{x},\vec{z}} tqr$ by precomposing with r and postcomposing with $s \oplus t$.

The existence of pairings in the last instance is immediate, so we are done. ◀

As such, given $\mathfrak{T}$ and a model M satisfying the conditions of Lemma 23, M has a well-defined interpretation of any $\approx$-polynomial-in-context $\Gamma \vdash p :: \Delta$ as the joint pairing of the monomials appearing in p (with the same context), followed by the codiagonal map.

▶ **Proposition 15.** *Let $\mathfrak{T}$ be a theory. There is an $\mathcal{A}$-natural equivalence of categories $\mathfrak{T}\text{-Mod}(\mathcal{A}) \simeq \mathbf{URing}(\mathbb{F}_1[\mathfrak{T}], \mathcal{A})$.*

Proof. We have thus far shown that a $\mathfrak{T}$-model M in $\mathcal{A}$ uniquely determines interpretations of $\approx^{\mathfrak{T}}$-polynomials-in-context. To produce a functor $\mathbb{F}_1[\mathfrak{T}] \to \mathcal{A}$, we need to extend this to $\approx^{\mathfrak{T}}$-matrices. Each entry of a matrix (p_{jk}) determines a morphism $M_0(T_k) \to M_0(U_j)$. Condition (†) of Definition 10 ensures that for fixed k, the interpretations of p_{jk} can be paired to give a morphism $M_0(T_k) \to \bigoplus_{j \in J} M_0(U_j)$; via the universal property of the coproduct, these produce a morphism $\bigoplus_{k \in K} M_0(T_k) \to \bigoplus_{j \in J} M_0(U_j)$. This extended construction ensures that the monoidal product and coproduct, as well as δ, δ^* and ϵ, are preserved.

Conversely, a morphism of ultrarings $F : \mathbb{F}_1[\mathfrak{T}] \to \mathcal{A}$ defines a model M by its restriction to the sorts and generators. The equations of $\mathfrak{T}$ necessarily hold because they correspond to morphisms forced to be equal in $\mathbb{F}_1[\mathfrak{T}]$, so are mapped by F to the equal morphisms in $\mathcal{A}$.

It is straightforward to verify that a model morphism uniquely determines a monoidal natural transformation, and conversely, since the latter are determined by their components at objects which generate the category under monoidal products and coproducts.

Finally, naturality in $\mathcal{A}$ follows from the fact that morphisms of ultrafunctors preserve the required structure: composing with an ultraring morphism $\mathcal{A} \to \mathcal{B}$ turns a $\mathfrak{T}$-model in $\mathcal{A}$ into one in $\mathcal{B}$. ◀

▶ **Theorem 17.** *Let $\mathcal{R}$ be an ultraring and $\mathfrak{T}$ its canonical theory, as defined in Definition 16. Then $\mathcal{R} \simeq \mathbb{F}_1[\mathfrak{T}]$.*

Proof. We have a functor $\ulcorner - \urcorner : \mathcal{R} \to \mathbb{F}_1[\mathfrak{T}]$ sending each object X to the sort $\ulcorner X \urcorner$ and each morphism $g : X \to Y$ to the corresponding generator (with no decomposition of X and Y as tensors or coproducts). We show that $\ulcorner - \urcorner$ is an equivalence of ultrarings.

Consider the generators (which we distinguish by a subscript) $\ulcorner \mathrm{id} \urcorner_1 : \ulcorner \bigotimes_{j \in J} A_j \urcorner \to \bigotimes_{j \in J} \ulcorner A_j \urcorner$ and $\ulcorner \mathrm{id} \urcorner_2 : \bigotimes_{j \in J} \ulcorner A_j \urcorner \to \ulcorner \bigotimes_{j \in J} A_j \urcorner$. By the second and first equations, these are mutual inverses in $\mathbb{F}_1[\mathfrak{T}]$. Moreover, naturality of these morphisms follows from the fourth axiom (applied inductively), so $\ulcorner - \urcorner$ preserves the monoidal product.

Consider the generator $\ulcorner \mathrm{id} \urcorner : \ulcorner \bigoplus_{j \in J} A_j \urcorner \to \bigoplus_{j \in J} \ulcorner A_j \urcorner$. By the third equation, we have that $\left(\ulcorner \mathrm{id} \urcorner^j (\vec{x}) = \vec{y}\right) \approx (\ulcorner \iota_j^* \urcorner (\vec{x}) = \vec{y})$, so this is equal to the row vector $(\ulcorner \iota_{j_1}^* \urcorner, \dots, \ulcorner \iota_{j_n}^* \urcorner)$.

Dually, we have generators $\ulcorner \iota_i \urcorner : \ulcorner A_i \urcorner \to \bigoplus_{j \in J} \ulcorner A_j \urcorner$. By the third, first and fifth equations, we have $\left(\ulcorner \iota_i \urcorner^j (\vec{x}) = \vec{y}\right) \approx \left(\ulcorner \iota_j^* \iota_i \urcorner (\vec{x}) = \vec{y}\right)$ which gives $(\vec{x} = \vec{y})$ if $i = j$ and 0 otherwise. Thus $\ulcorner - \urcorner$ preserves the coproduct injections; in particular, these morphisms are compatible, so we can construct the column vector $(\ulcorner \iota_{j_1} \urcorner, \dots, \ulcorner \iota_{j_n} \urcorner)^t$. We can use the second and third axioms to conclude that this column vector is the inverse to $\ulcorner \mathrm{id} \urcorner$, so $\ulcorner - \urcorner$ preserves coproducts. Formally, we must also check that the universal morphisms are respected by this isomorphism, in the sense that $\int_{\vec{y}} (\ulcorner [f_1, \dots, f_n] \urcorner (\vec{x}) = \vec{y})\, (\ulcorner \mathrm{id} \urcorner (\vec{y}) = \vec{z}) \approx \int_{\vec{y}'} (\ulcorner \mathrm{id} \urcorner (\vec{x}) = \vec{y}')\, ((\ulcorner f_1 \urcorner, \dots, \ulcorner f_n \urcorner)\, (\vec{y}') = \vec{z})$. It suffices to check this componentwise by composing with ι_i, which works by the preceding observations and the second axiom.

For the δ^*-category structure, we observe that the sixth, seventh and eighth axioms directly ensure that these are preserved up to the isomorphisms constructed above. With this, we have shown that $\ulcorner - \urcorner$ is a valid morphism of ultrarings.

Conversely, there is a canonical model of the canonical theory $\mathfrak{T}$ in $\mathcal{R}$, given by interpreting $\ulcorner A \urcorner$ as A and $\ulcorner g \urcorner$ as g (with the relevant typing). The claim that this $\mathfrak{S}$-structure is indeed a model amounts to saying that the interpretations of the monomials appearing in the axioms of Definition 16 are equal in $\mathcal{R}$, which is straightforward to check.

Finally, to see that $\ulcorner - \urcorner$ is an equivalence, we simply observe on the one hand that the induced model in $\mathbb{F}_1[\mathfrak{T}]$ is isomorphic to that corresponding to the identity functor and on the other that the induced endofunctor of $\mathcal{R}$ is naturally isomorphic to the identity. The involved natural isomorphisms are the structural ones constructed above. ◀

A.3 Harmless Predicates Over the Real Numbers

▶ **Lemma 24.** *In the case $\mathbb{K} = \mathbb{R}$, the truth value of a harmless predicate may be computed in constant time on a $\mathbb{R}$-RAM.*

Proof. For each harmless predicate P we have an equation of the form $P(t)\varphi(t) \approx_\Gamma 0$, so for each t, to know whether $P(t)$ is true in a given model in $\mathbb{R}$ we just need to check whether the interpretation of $\varphi(t)$ has a root in $\mathbb{R}$. The idea here is to reduce this problem as a series of positivity tests that can be done in constant time by generalizing the idea of the discriminant for n variate polynomials of degree two. Let us give some details about this generalization.

If the polynomial is of degree zero the test is trivial because it has no roots. If it is of degree one, since every polynomial of $\mathbb{R}$ of odd degree have a root in $\mathbb{R}$, we know that there is a root in $\mathbb{R}$. The only non trivial case is the degree two. Since we work in $\mathbb{R}$ the polynomial has a root in $\mathbb{R}$ iff it is reducible. First, let us do the special case of an homogeneous polynomial. A quadratic form q can be written as a matrix multiplication $x^T Ax$, where A is the Kronecker matrix of the quadratic form and is a symmetric matrix. We now show that A has rank one iff q is reducible. First if q is reducible then we can write the q as $(\sum_i b_i x_i)(\sum_i c_i x_i)$, by rewriting it with matrices multiplication $q = x^T \begin{pmatrix} b_0 & & \\ & \cdots & \\ & & b_n \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} c_0 & & \\ & \cdots & \\ & & c_n \end{pmatrix} x$ so A is of rank 1. Conversely if A is of rank one, since we know that A is a real symmetric matrix, it is diagonalizable. Hence, A is similar to $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ which is also similar to $\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$ thus we can rewrite q as a product of two degree one polynomials. Moreover A is of rank one iff it has exactly one non-zero eigenvalue. Hence, we need 0 to be root of its characteristic polynomial with multiplicity $m-1$, m being the number of variables, we can then rewrite the general form of the characteristic polynomial as $\lambda^{m-1}((-1)^m \lambda + (-1)^{m-1}\mathrm{Tr}(A))$. By identifying with the formula $\det(A - \lambda I_m)$ we are given a set of constraint on the coefficients of the quadratic form for it to be reducible. Let us now consider a general multivariate polynomial of degree two. We can write it as $p = \sum_{i,j} a_{i,j} x_i x_j + \sum_i a'_i x_i + \sum_i a''_i$ the x_i being variables taken in a set X. The following polynomial $q = \sum_{i,j} a_{i,j} x_i x_j + y(\sum_i a'_i x_i) + y^2(\sum_i a''_i)$ is a quadratic form. We need to show that q is reducible iff p is. This comes from the fact that if we note $x' := (x_0, \cdots, x_m, 1)$ we have $p = x' A x'^T$ where A is the Kronecker matrix of q. To go from a factorisation of q to one of p is simply to replace y by 1 and the other way is only multiplying constant terms of each factor by y.

To conclude, we need to show that these constraints we exhibited are fixed by the theory $\mathfrak{T}$ and can be hardcoded as a sequence into the code of the machine for each harmless predicate in $\mathfrak{T}$. This means the machine doesn't have to compute the characteristic polynomial each time but only compute the set of constraint we can deduce of it, this can be done in constant time. Since the polynomial might depend on non-harmless generators they should all be computed by the machine before the harmless predicate, which can be done because, by definition, non-harmless generators only depends on $t-1$ generators. ◀