Undecidability for Semirings with Fixed Points

Anupam Das
University of Birmingham, UK

Abhishek De
Krea University, India

Stepan L. Kuznetsov
Steklov Mathematical Institute of RAS, Russia

Abstract

In this work, we prove the undecidability (and Σ^0_1-completeness) of several theories of semirings with fixed points. The generality of our results stems from recursion theoretic methods, namely the technique of *effective inseparability*. Our result applies to many theories proposed in the literature, including Conway μ-semirings, Park μ-semirings, and Chomsky algebras.

2012 ACM Subject Classification Theory of computation → Grammars and context-free languages; Theory of computation → Algebraic language theory; Theory of computation → Logic and verification

Keywords and phrases Semirings, fixed points, decidability, context-free, inseparability

Digital Object Identifier 10.4230/LIPIcs.FSCD.2026.15

Funding *Anupam Das*: This author has been supported by a UKRI Future Leaders Fellowship, Structure vs Invariants in Proofs, project number MR/S035540/1.
Abhishek De: This author has been supported by a UKRI Future Leaders Fellowship, Structure vs Invariants in Proofs, project number MR/S035540/1.
Stepan L. Kuznetsov: The work of this author was performed at the Steklov International Mathematical Center and supported by the Ministry of Science and Higher Education of the Russian Federation (agreement no. 075-15-2025-303).

1 Introduction

The *context-free languages* (CFLs) may be presented as the least solutions to certain systems of algebraic inequalities, by construing *context-free grammars* (CFGs) as such. It is well known that the resolution of such simultaneous fixed points may be reduced to unary ones, by *Bekić's Lemma*, [1]. Indeed, already in the early '70s, Gruska observed that expressions for CFLs could be obtained by extending usual regular expressions by an "iteration" operator for computing (least) fixed points. Nowadays these are known as *μ-expressions* [20, 6, 7].[1]

Algebraic theories of regular expressions are based on (idempotent) semirings with an appropriate $*$ operation, for instance *Kleene algebras* (cf., e.g., [14]). Duly, analogous structures for μ-expressions are based on (idempotent) semirings with *fixed points*. Similar to regular expressions, in the last few decades several formulations of semirings with fixed points have appeared throughout the literature in formal language theory, logic, and algebra [2, 20, 6, 7, 12, 13, 5, 8, 21].

However, until now, as far as we know, the (un)decidability of most of these equational theories has remained open. This is in stark contrast to the situation for Kleene algebras, over regular expressions, where the complexity of equational and Horn theories, with and without continuity, is now well understood (see, e.g., [16, 4]). In this work, we address this

[1] Here the notation "μ" emphasises that the fixed points named in the syntax are *least*.

situation by showing undecidability (in fact Σ^0_1-completeness) for a range of theories of (not necessarily continuous) semirings with fixed points, including many of the ones occurring in the literature, in particular from [20, 6, 7, 8].

1.1 Previous work

Leiß investigated several classes of idempotent semirings with fixed points, all of which satisfy the *induction axioms* characterising $\mu x\, e(x)$ as the least prefixed point of the operation $x \mapsto e(x)$. This interpretation of least fixed points is recovered from the famous Knaster-Tarski fixed point theorem, and is the standard way to present axiomatic theories with forms of induction. Ésik and Leiß showed that reduction of CFGs to Greibach normal form [9] can be validated in a purely equational theory of semirings with fixed points (not necessarily satisfying induction), *Conway μ-semirings* [6, 7].

Grathwohl, Henglein and Kozen gave a sound and complete (infinitary) axiomatisation of CFL inclusions by way of μ-expressions [8] (note that this problem is Π^0_1-complete). In particular, they introduced the notion of a *Chomsky algebra*, an algebraically closed idempotent semiring. These turn out to be nothing more than idempotent Park μ-semirings of [20, 6, 7], where μ-expressions give a naming convention for solutions to polynomial systems. In [8], it is shown that the theory of *μ-continuous* Chomsky algebras is sound and complete for CFL inclusions.

Later, further metalogical results were established for Chomsky algebras and friends. For instance, Leiß showed that the matrix ring of a μ-continuous Chomsky algebra is again μ-continuous. Two of the authors of the present work recovered the completeness result of [8] via proof-theoretic means, even establishing a stronger "cut-free" completeness result [3]. In the same work, they also proposed *non-wellfounded* systems for both CFLs, via μ-expressions, and an extension to ω-words, using *greatest fixed points*.

1.2 Contributions

In this work, we show undecidability, in fact Σ^0_1-completeness, for a range of theories of semirings with fixed points. We give a basic equational axiomatisation $\mu\mathsf{S}_\chi$ of semirings (not necessarily with identity) with a μ operator denoting (not necessarily least) fixed points. In particular, this theory is more general (i.e. weaker) than the Conway μ-semirings, Park μ-semirings and Chomsky algebras from earlier works. Our main result is that *any* recursively enumerable extension of $\mu\mathsf{S}_\chi$ sound for the usual language model is undecidable, and in fact Σ^0_1-complete. In particular, Conway μ-semirings, Park μ-semirings, and Chomsky algebras have undecidable equational theories.

To achieve such generality, we employ methods from recursion theory, namely *effective inseparability*. These methods were successfully employed in earlier works by one of the authors, in particular proving undecidability of the equational theory of action lattices [17, 18], a longstanding open problem, and the same for commutative action lattices [19].

1.3 Structure of the paper

In Section 2, we introduce a basic theory $\mu\mathsf{S}$ of semirings (with identity) with fixed points, as well as other theories from the literature. In Section 3, we prove the effective inseparability of two problems on CFGs, "non-totality" $\bar{\mathbf{T}}$ and "regular totality" $\mathbf{R}$, by reduction to a more well-known effectively inseparable pair, "halting" and "looping" Turing machines. In Section 4, we show the undecidability of any r.e. extension of $\mu\mathsf{S} + \mathit{idempotency}$ (i.e. $e + e = e$)

that is sound for the language model. We extend this result to theories without identity and idempotency in Section 5 by interpreting idempotency using fixed points. Finally, we make some concluding remarks in Section 6.

2 Some theories of semirings with fixed points

Let $\mathcal{A}$ be a finite alphabet and $\mathcal{V}$ be a countably infinite set of variables disjoint from $\mathcal{A}$. We consider **μ-expressions** e, f, etc. generated by the following grammar:

$$e, f, \ldots \quad ::= \quad 0 \quad | \quad 1 \quad | \quad a \in \mathcal{A} \quad | \quad X \in \mathcal{V} \quad | \quad e + f \quad | \quad e \cdot f \quad | \quad \mu X e$$

The notion of **free variable** is defined as expected, construing μ as a variable binder. We usually just write ef instead of $e \cdot f$.

Structures interpreting μ-expressions, formally, must be defined appropriately, taking account of variable binding and compatible with substitution. A formal exposition can be found in e.g. [7, 21], so here we give only examples:

▶ **Example 1** (Languages). Temporarily expand the syntax of expressions to include each language $A \subseteq \mathcal{A}^*$ as a new constant symbol. We associate to each closed μ-expression e (of this expanded syntax) a language $\mathcal{L}(e) \subseteq \mathcal{A}^*$ as follows:

$$\mathcal{L}(0) := \varnothing \qquad \mathcal{L}(a) := \{a\} \qquad \mathcal{L}(e+f) := \mathcal{L}(e) \cup \mathcal{L}(f)$$
$$\mathcal{L}(1) := \{\varepsilon\} \qquad \mathcal{L}(A) := A \qquad \mathcal{L}(ef) := \{uv : u \in \mathcal{L}(e) \text{ and } v \in \mathcal{L}(f)\}$$
$$\mathcal{L}(\mu X e(X)) := \bigcap\{A : \mathcal{L}(e(A)) \subseteq A\}$$

Every symbol is interpreted by $\mathcal{L}(\cdot)$ as a monotone operation, wrt. $(\mathcal{P}(\mathcal{A}^*), \subseteq)$. The *Knaster-Tarski* fixed point theorem tells us that monotone operations on complete lattices have least (and greatest) fixed points, in particular with the least fixed point of $A \mapsto e(A)$ given by $\mathcal{L}(\mu X e(X))$ defined above: the intersection of all *prefixed points* of $A \mapsto e(A)$.

It is not hard to see that the image of $\mathcal{L}(\cdot)$ on μ-expressions is just the context-free languages (CFLs) [10]. This structure will form a model of all theories considered herein, and will guide our main undecidability results. Let us point out already the following consequence:

▶ **Proposition 2** (See, e.g., [11, 24]). *Deciding $\mathcal{L}(e) \subseteq_? \mathcal{L}(f)$ is Π^0_1-complete.*

2.1 Semirings with fixed points

For generality, we work with a minimal theory of semirings with fixed points, before considering specific examples such as (idempotent) Park μ-semirings and Conway μ-semirings later.

▶ **Definition 3** (Basic axioms). *The equational theory $\mu\mathsf{S}$ of μ-expressions is axiomatised by:*

(i) *$(0, 1, +, \cdot)$ form a semiring:*

$$e + (f + g) = (e + f) + g \qquad e(fg) = (ef)g \qquad e(f+g) = ef + eg$$
$$e + f = f + e \qquad e1 = e = 1e \qquad (e+f)g = eg + fg$$
$$e + 0 = e \qquad e0 = 0 = 0e$$

(ii) *$\mu X e(X)$ is a fixed point of $X \mapsto e(X)$: $\mu X e(X) = e(\mu X e(X))$.*[2]

[2] We shall typically assume, when writing $\mu X e(X)$, that all free occurrences of X in $e(X)$ have been distinguished.

(iii) *α-equivalence: $\mu X e(X) = \mu Y e(Y)$, as long as Y is not free in $e(X)$.*

Note that $\mu\mathsf{S}$ does *not* insist that $\mu X e(X)$ is a *least* fixed point, which would require non-equational axioms. Instead, notice that the axiomatisation above is finite and equational.

▶ Remark 4 (On μ-congruence). Note that $\mu\mathsf{S}$ does not admit the ***μ*-congruence** rule:

$$\frac{e(X) = f(X)}{\mu X e(X) = \mu X f(X)}$$

This means that, even if two operators are equivalent, the fixed points named by their closure under μ may be different. For instance $\mu\mathsf{S} \not\vdash \mu X(a+X) = \mu X(X+a)$. This is consistent with the fact that we are not axiomatising $\mu X e(X)$ to be a *least* fixed point, but just *some* fixed point. So, e.g., we can tweak the model $\mathcal{L}$ to satisfy $\mu X(a+X) = a$ but $\mu X(X+a) = a+b$. Only the former is the *least* fixed point, but the resulting structure remains a model of $\mu\mathsf{S}$.

Construing expressions in a higher-order syntax, where μ is a higher-order constant rather than a binder, the lack of μ-congruence may be seen as saying that μ does not behave *extensionally*. In practice, in this work, this means we must be careful not to carry out reasoning *under* a μ unless justified.

▶ **Definition 5** (μ-semirings). *A **μ-semiring** (e.g., [7, 8, 21]) is defined just like $\mu\mathsf{S}$ except:*

- *μ-semirings must satisfy μ-congruence (cf. Remark 4 above); and,*
- *μ-semirings do not require $\mu X e(X)$ to be a fixed point, i.e. axiom* (ii) *is omitted.*

Our system $\mu\mathsf{S}$ is, strictly speaking, incomparable with general μ-semirings, cf. Remark 4, though it will still underlie extensions considered later. $\mu\mathsf{S}$ should rather be considered a basic theory of "semirings with fixed points", namely the theory of those μ-semirings in which $\mu X e(X)$ always denotes a fixed point of $X \mapsto e(X)$, albeit not necessarily the *least* fixed point, and albeit not extensionally so (see previous Remark 4).

We omit μ-congruence for two main reasons: (a) we do not need it to obtain our ultimate undecidability results, rendering them strictly more general; and (b) the axiomatisation $\mu\mathsf{S}$ remains purely equational, and so could be of independent interest. We shall soon present μ-semirings equipped with an order, to axiomatise the *leastness* of fixed points denoted by μ, in particular satisfying the "strong" congruence principle mentioned in the example above. Before that, let us recall a class of models of $\mu\mathsf{S}$ that is covered by our results:

▶ **Definition 6** ([7]). *A **Conway μ-semiring** is a μ-semiring further satisfying:*

1. $\mu X e(f(X)) = e(\mu X f(e(X)))$
2. $\mu X \mu Y e(X,Y) = \mu X e(X,X)$

Note that the fixed point axiom (ii) is already a consequence of (1) above, by setting $f(X) := X$, and so Conway μ-semirings are also models of $\mu\mathsf{S}$. Conway μ-semirings were introduced in [7] as a minimal theory in which certain normal forms of context-free grammars may be formally obtained, in particular *Greibach* normal form. We shall not develop its metatheory any further here, as our results about Conway μ-semirings depend only on the fact that their equational theory is recursively enumerable, extends that of $\mu\mathsf{S}$, and is modelled by the structure of languages $\mathcal{L}$ from Example 1.

Before proceeding to give further variations of μ-semirings, it is illustrative to point out that, without μ, the theory of semirings is indeed decidable. This is surely a folklore result, but we develop it in Appendix A for the sake of completeness.

2.2 Equational systems and context-free grammars

Before considering further extensions of $\mu\mathsf{S}$, let us revisit the interpretation of μ-expressions as context-free languages. It is known that Conway μ-semirings can "solve" context-free grammars (CFGs), but we shall recast (and improve) this result within $\mu\mathsf{S}$ here.

An **equational system** is a set $\mathcal{E}(\vec{X}) = \{X_i = f_i(\vec{X})\}_{i<n}$. By recursively applying *Bekić's Lemma* [1] we can solve such systems in $\mu\mathsf{S}$:

▶ **Proposition 7.** *Fix an equational system $\mathcal{E}(\vec{X}) = \{X_i = f_i(\vec{X})\}_{i<n}$. There are (closed) expressions $\vec{e}$ s.t. $\mu\mathsf{S} \vdash \mathcal{E}(\vec{e})$.*

Proof. By induction on n:

- When $n = 0$, the statement is vacuously true.
- Fix an equational system $\{X_i = f_i(\vec{X}, X_n)\}_{i\leq n}$, where $\vec{X} = X_0, \ldots, X_{n-1}$. Write $e'_n(\vec{X}) := \mu X_n f_n(\vec{X}, X_n)$. Now, by inductive hypothesis, let $\vec{e}$ be solutions of $\{X_i = f_i(\vec{X}, e'_n(\vec{X}))\}_{i<n}$, i.e. such that $\mu\mathsf{S} \vdash e_i = f_i(\vec{e}, e'_n(\vec{e}))$, for all $i < n$. It suffices now to show that $e_n := e'_n(\vec{e})$ is a solution for X_n, i.e. $\mu\mathsf{S} \vdash e_n = f_n(\vec{e}, e_n)$, for which we have:

$$\begin{aligned} e_n &= e'_n(\vec{e}) && \text{by definition of } e_n \\ &= \mu X_n f_n(\vec{e}, X_n) && \text{by definition of } e'_n \\ &= f_n(\vec{e}, \mu X_n f_n(\vec{e}, X_n)) && \text{by fixed point axiom (ii)} \\ &= f_n(\vec{e}, e_n) && \text{by definition of } e_n \end{aligned}$$

◀

Recalling Remark 4, note that solutions to an equational system are not necessarily unique, as $\mu\mathsf{S}$ does not insist that μ computes *least* fixed points.

Of course, we can use the above result to "solve" CFGs too.

▶ **Corollary 8.** *Fix a CFG $\{X_i \Rightarrow f_{ij}(\vec{X})\}_{i<n,j}$. There are μ-expressions $\vec{e} = e_0, \ldots, e_{n-1}$ s.t. $\mu\mathsf{S} \vdash e_i = \sum_j f_{ij}(\vec{e})$ for all $i < n$.*

2.3 Ordered semirings with fixed points

Let us continue to recall certain classes of semirings over μ-expressions from the literature.

A semiring is **idempotent** if it satisfies $\mathsf{idem} : e + e = e$. We write $e \leq f := e + f = f$, typically in the presence of idempotency. Note $\leq$, called the "natural order", is a partial order under $\mu\mathsf{S} + \mathsf{idem}$, and moreover, one over which the semiring operations are monotone:

▶ **Proposition 9.** *The following hold in all idempotent semirings, and so in particular are consequences of $\mu\mathsf{S} + \mathsf{idem}$, for all expressions e, f, g, i:*

$$\begin{array}{ll} i \leq i & e \leq f \implies e + i \leq f + i \\ e \leq f \;\&\; f \leq e \implies e = f & e \leq f \implies eg \leq fg \\ e \leq f \;\&\; f \leq g \implies e \leq g & e \leq f \implies ge \leq gf \end{array} \tag{1}$$

Moreover these properties hold in any *semiring whenever i is idempotent, and in particular are consequences of $\mu\mathsf{S} + (i + i = i)$.*

▶ **Definition 10** (Ordered μ-semirings (see, e.g., [5, 6])). *An **ordered μ-semiring** is a μ-semiring equipped with a partial order satisfying the properties from Equation* (1) *above, as well as an ordered analogue of μ-congruence, **μ-monotonicity**:*

$$\frac{e(X) \leq f(X)}{\mu X e(X) \leq \mu X f(X)}$$

Again, μ-monotonicity does not necessarily hold in models of $\mu\mathsf{S}+\mathsf{idem}$, even ones closed under μ-congruence. E.g. if $e(X)$ is $a+X$ and $f(X)$ is $a+b+X$, then $e(X) \leq f(X)$ is valid in all models of $\mu\mathsf{S}+\mathsf{idem}$, but $\mu X e(X)$ needs only be greater than a and $\mu X f(X)$ needs only be greater than $a+b$. Again, tweaking the language model, it is consistent to interpret $\mu X e(X)$ as $\{a,b,c\}$ but $\mu X f(X)$ as $\{a,b\}$. Note, since $e(X) \leq f(X)$ is valid in *all* models of $\mu\mathsf{S}+\mathsf{idem}+\mu$-congruence, μ-monotonicity is not even admissible for this theory.

In this work, we shall only consider the "natural order" already defined, $e \leq f := e+f = f$, before more generally structures without idempotency. It is worth recalling at this point:

▶ **Fact 11.** *The set of all idempotent elements of a semiring forms an ideal.*

In the presence of an order, it is of particular interest to examine structures where μ calculates a *least* (pre)fixed point wrt. the order:

$$\mathsf{lfp}: \quad e(\mu X e(X)) \leq \mu X e(X) \qquad \& \qquad e(f) \leq f \implies \mu X e(X) \leq f$$

Note that the axioms above are neither (in)equational nor finitely many: there is an axiom for each choice of f above.

▶ **Definition 12.** *A **Park μ-semiring** is an ordered μ-semiring satisfying* lfp*. A **Chomsky Algebra** is an idempotent Park μ-semiring whose order is the natural order.*[3]

Chomsky Algebras were defined in [8, 21] as *algebraically complete* idempotent semirings, while Park μ-semirings were defined axiomatically in [6, 7]. Both presentations are equivalent when construing μ-expressions as a naming convention for equational systems (see previous subsection), as was noticed already in [8]. Note that the fixed point axiom (ii) is already a consequence of lfp (see [6, 7]), and so Chomsky Algebras are models of $\mu\mathsf{S}+\mathsf{idem}$, and Park μ-semirings are models of $\mu\mathsf{S}$.

Again, we shall not recount the established metatheory of (idempotent) Park μ-semirings or Chomsky Algebras: what is important for our results are that they are recursively enumerable extensions of $\mu\mathsf{S}(+\mathsf{idem})$ that are still sound for $\mathcal{L}$.

3 Effective inseparability of two problems on CFGs

Before our main results on the undecidability of various theories of (idempotent) semirings with fixed points, we take a detour through the recursion theory of problems on CFGs.

Recall that Σ^0_1 is the class of all recursively enumerable (r.e.) sets, and Π^0_1 is the class of co-r.e. sets, i.e. those sets whose complements are r.e.

3.1 Two problems on CFGs

For our arguments, we use *effective inseparability* of two algorithmic problems for context-free grammars (CFG). We consider only a special class of CFGs, which we call *productive* ones. Let $\mathcal{A}$ be the set of all terminal symbols and $\mathcal{V}$ be the set of all non-terminal ones. As usual, we shall assume among the non-terminals a designated **start** symbol $S \in \mathcal{V}$.

▶ **Definition 13** (Productive CFGs)**.** *A CFG is **productive** if, for every production rule $X \Rightarrow \alpha_1 \dots \alpha_n$, some α_i is a terminal, i.e. some $\alpha_i \in \mathcal{A}$.*

[3] In fact, the final clause here is redundant: the *only* monotone partial order on an idempotent semiring is the natural order.

A typical example of a productive CFG is a Greibach grammar without ε-productions. In Greibach grammars, we always have $\alpha_1 \in \mathcal{A}$. Thus, by Greibach's normalisation theorem [9], one shows that any context-free language without the empty word is generated by a productive CFG. On the other hand, the empty word cannot be generated by such a grammar. In the sequel, we shall restrict our attention to productive CFGs.

The first problem we consider is the *totality* problem (i.e., the question whether a given productive CFG generates all nonempty words over $\mathcal{A}$). The second one is more specific. It could be called "the grammar is total for obvious reasons," or *regularly total*. Formally:

▶ **Definition 14** ((Regular) totality). *A productive CFG is* ***total****, if it generates (from S) all nonempty words over $\mathcal{A}$. A productive CFG G is* ***regularly total****, if it includes two designated non-terminals $Y, T \in \mathcal{V}$ such that:*

- *For each $a \in \mathcal{A}$, G includes all productions of the form $S \Rightarrow aYT$, $T \Rightarrow a$ and $T \Rightarrow aT$.*
- *There is some $n > 0$ such that:*
  1. *S generates all nonempty words of length $\leq n+1$; and,*
  2. *Y generates all words of length n.*

It is not hard to see that every regulary total productive CFG is total:

▶ **Proposition 15** (Regular totality implies totality). *Regularly total productive CFGs are total.*

Proof. Let G be a regularly total productive CFG and let $\vec{a}$ be a nonempty word:

- If $|\vec{a}| \leq n+1$ then it is generated from S by definition of regular totality.
- If $|\vec{a}| > n+1$ then write $\vec{a} = a\vec{b}\vec{c}$, for some $a \in \mathcal{A}$, $|\vec{b}| = n$ and $|\vec{c}| > 0$. Then $S \Rightarrow aYT$ and Y generates $\vec{b}$, by assumption, and T generates $\vec{c}$ (since it trivially generates *every* nonempty word). ◀

The algorithmic complexity of regular totality is dual to that of totality: while the latter is known to be Π^0_1-complete, the former belongs to Σ^0_1 (and, as we shall see below, is also complete in this class).

In what follows, we tacitly identify productive CFGs with their Gödel numbers under a reasonable encoding. Write $\mathbf{R}$ for the set of (Gödel numbers of) regularly total productive CFGs and $\overline{\mathbf{T}}$ for the set of (Gödel numbers of) non-total productive CFGs.

3.2 Effective inseparability by reduction

We recall, following [18], the notion of effective inseparability and the results which allow it to be used for proving Σ^0_1-completeness of logical theories.

Fix some acceptable enumeration $\{f_n\}_{n\in\mathbb{N}}$ of the one-input partial recursive functions on $\mathbb{N}$, and let us write W_n for "the n^{th} r.e. set," i.e. $\mathsf{dom}(f_n)$, the domain of f_n.

▶ **Definition 16.** *$A, B \subseteq \mathbb{N}$ are* ***effectively inseparable*** *if:*

- *$A \cap B = \varnothing$; and,*
- *there is a partial recursive function $h(x,y)$ s.t., whenever $W_u \supseteq A$ and $W_v \supseteq B$ with $W_u \cap W_v = \varnothing$, we have $h(u,v)$ is defined and $h(u,v) \notin W_u \cup W_v$.*

We shall call such $h(x,y)$ a ***productive function*** *for A, B.*

Given disjoint $A, B \subseteq \mathbb{N}$, a set $C \subseteq \mathbb{N}$ **separates** A from B (or is a **separator**) if $C \supseteq A$ but nonetheless $C \cap B = \varnothing$. In the Definition above, informally, h uniformly falsifies any attempt to separate A from B by a decidable set. Intuitively, we can see $\lambda v\, h(u,v)$ here as a "proof" that any such W_u is not decidable: it is a recursive function that computes counterexamples to the statement that W_u has an r.e. complement.

The next result is a corollary of a theorem by Myhill [22], as shown in [23, Exercise 11-14]:

▶ **Theorem 17.** *If $A, B \subseteq \mathbb{N}$ are effectively inseparable r.e. sets and C is an r.e. set separating them, then C is Σ^0_1-complete.*

Note in the Theorem above, and generally in the sequel, we shall not be precise about whether C separates A from B or vice-versa, WLoG by symmetry of the statement.

Effective inseparability can be propagated by a (very weak) kind of reduction. Given $A, B \subseteq \mathbb{N}$ with $A \cap B = \varnothing$ and $A', B' \subseteq \mathbb{N}$ with $A' \cap B' = \varnothing$, an **e.i.-reduction** from (A', B') to (A, B) is a total computable function f such that:

1. $f(A) \subseteq A'$
2. $f(B) \subseteq B'$

Such maps are indeed appropriate for reducing questions of effective inseparability:

▶ **Proposition 18.** *Suppose $A, B \subseteq \mathbb{N}$ are effectively inseparable, and $A', B' \subseteq \mathbb{N}$ with $A' \cap B' = \varnothing$. If f is an e.i.-reduction from (A', B') to (A, B) then A' and B' are also effectively inseparable.*

Proof. The preimage of an r.e. set under a total computable mapping is itself r.e. Indeed we have $f^{-1}(W_n) = \mathsf{dom}(f_n \circ f)$, and we can even effectively compute from n the index of the r.e. function $f_n \circ f$, say by a total recursive function g, so that always $f^{-1}(W_n) = W_{g(n)}$.

Now, let h be a productive function for A, B and let us define a productive function h' for A', B' as follows:

$$h'(u, v) = f(h(g(u), g(v)))$$

Clearly h' is a partial recursive function. To show it is productive for A', B', suppose $W_u \supseteq A'$, $W_v \supseteq B'$ with $W_u \cap W_v = \varnothing$ and let us show that $h'(u, v) \notin W_u \cup W_v$. First, we have:

$$\begin{array}{rcll} W_{g(u)} & = & f^{-1}(W_u) & \text{by definition of } g \\ & \supseteq & f^{-1}(A') & \text{since } W_u \supseteq A' \\ & \supseteq & A & \text{since } f(A) \subseteq A' \end{array} \qquad \begin{array}{rcll} W_{g(v)} & = & f^{-1}(W_v) & \text{by definition of } g \\ & \supseteq & f^{-1}(B') & \text{since } W_v \supseteq B' \\ & \supseteq & B & \text{since } f(B) \subseteq B' \end{array}$$

$$\begin{array}{rcll} W_{g(u)} \cap W_{g(v)} & = & f^{-1}(W_u) \cap f^{-1}(W_v) & \text{by definition of } g \\ & = & f^{-1}(W_u \cap W_v) & \text{since preimages preserve intersections} \\ & = & f^{-1}(\varnothing) & \text{since } W_u \cap W_v = \varnothing \\ & = & \varnothing & \end{array}$$

From here we have:

$$\begin{array}{rcll} h(g(u), g(v)) & \notin & W_{g(u)} \cup W_{g(v)} & \text{since } h \text{ is productive for } A, B \\ h(g(u), g(v)) & \notin & f^{-1}(W_u) \cup f^{-1}(W_v) & \text{by definition of } g \\ h(g(u), g(v)) & \notin & f^{-1}(W_u \cup W_v) & \text{since preimages preserve unions} \end{array}$$

Hence $h'(u, v) = f(h(g(u), g(v)) \notin W_u \cup W_v$, as required. ◀

3.3 Effective inseperability of $\mathbf{R}$ and $\bar{\mathbf{T}}$

This subsection is devoted to proving the following result.

▶ **Theorem 19.** *The sets $\mathbf{R}$ and $\bar{\mathbf{T}}$ are effectively inseparable.*

We proceed via a reduction from a more well-known pair of effectively inseparable problems: halting and looping for Turing machines. Throughout this section we shall work with one-input deterministic Turing machines that include, besides the initial and final state, also a designated *looping* state q_c with rules that force the machine to remain at this state once it is reached: $(q_c, a) \Rightarrow (q_c, a, N)$ for any letter a of the tape alphabet.

We say that a machine $\mathtt{M}$ loops on an input x if $\mathtt{M}$ reaches q_c when running on x. Looping is a very specific case of non-halting. Let

$$\mathbf{C} = \{(\mathtt{M}, x) \mid \mathtt{M} \text{ loops on input } x\} \qquad \text{and} \qquad \mathbf{H} = \{(\mathtt{M}, x) \mid \mathtt{M} \text{ halts on input } x\}$$

Again, we construe both these sets as subsets of $\mathbb{N}$ by implicitly identifying pairs $(\mathtt{M}, x)$ with their codes. The following fact is folklore (see, e.g., [23, Exercise 7-55d][18, Proposition 3]):

▶ **Proposition 20.** *The sets* $\mathbf{C}$ *and* $\mathbf{H}$ *are effectively inseparable.*

Now we can prove the effective inseparability of $\mathbf{R}$ and $\bar{\mathbf{T}}$ by reduction, in the sense of Proposition 18, to the effective inseparability of $\mathbf{C}$ and $\mathbf{H}$.

Let us start by briefly recalling some ideas from the standard proof of Π^0_1-hardness for the totality problem for CFGs, as presented in Kozen's textbook [15]. The *halting protocol* of a Turing machine $\mathtt{M}$ on a given input x (if it exists) is the word $\#\kappa_0\#\kappa_1\#\ldots\#\kappa_n\#$, where κ_0 is the initial configuration q_0x, each κ_{i+1} is the configuration which is the immediate successor of κ_i, and κ_n is a final configuration. Let $\mathcal{A}$ be the alphabet in which configurations are written down (which includes the internal alphabet of $\mathtt{M}$, its set of states, and the additional symbol #). We may readily define a (computable) reducing function, mapping each pair $(\mathtt{M}, x)$ to a CFG which generates all non-empty words except for the halting protocol (if it exists). Thus, this grammar is total if and only if $\mathtt{M}$ does not halt on x.

We shall modify this construction a bit, following [18], in order to take care of looping. All non-empty words which are not the halting protocol must belong to one of the following three classes:

- **(I)** words of length greater than 1, beginning with #, which cannot be even a prefix of the halting protocol: this includes words not beginning with $\#\kappa_0\#$; words where there is not a correct configuration between two #'s; words with subwords $\#\kappa\#\kappa'\#$ where κ' is not the immediate successor of κ; and, most importantly, all words including q_c;
- **(II)** words of length greater than 1, beginning with #, and not ending with $\#\kappa_F\#$ where κ_F is a final configuration (such a word could be an incomplete protocol – a prefix of the halting protocol);
- **(III)** words beginning with a letter different from #, and the one-letter word #.

Note that these classes are not necessarily disjoint: a word can belong to more than one of them.

Write (I)′ and (II)′ for the sets of words from (I) and (II), respectively, with the first symbol # removed. The standard construction shows that both (I)′ and (II)′ are context-free. Denote the grammars generating (I)′ and (II)′ by G_1 and G_2, respectively; we suppose that both grammars are productive (e.g., being Greibach grammars). Here it is important that (I)′ and (II)′ are ε-free. Moreover, G_1 and G_2 are computable from $(\mathtt{M}, x)$ [18, Section 2.1].

We assume G_1 and G_2 have disjoint sets of non-terminals, and write Y and Z for their start symbols, respectively. Now define the grammar $G_{\mathtt{M},x}$ obtained from the union of G_1 and G_2 along with the following additional productions:

$$
\begin{array}{lll}
S \Rightarrow aYT & \text{for each } a \in \mathcal{A} & S \Rightarrow \#Y \\
T \Rightarrow aT & \text{for each } a \in \mathcal{A} & S \Rightarrow \#Z \\
T \Rightarrow a & \text{for each } a \in \mathcal{A} & S \Rightarrow aT \quad \text{for each } a \in \mathcal{A} - \{\#\} \\
 & & S \Rightarrow a \quad \text{for each } a \in \mathcal{A}
\end{array}
$$

Here S and T are fresh non-terminals, and S is the new start symbol of $G_{\mathtt{M},x}$. Notice that the first three productions are exactly as in Definition 14 for regular totality.

▶ **Proposition 21.** *$G_{\mathtt{M},x}$ generates all nonempty words over $\mathcal{A}$, except the halting protocol of $\mathtt{M}$ on x (if it exists).*

Proof. For words from (I) or (II) we start by firing $S \Rightarrow \#Y$ and $S \Rightarrow \#Z$, respectively, and continue as in G_1 or G_2, respectively. Words from (III) are generated by the final two rules. Note that this analysis accounts for *all* words generated by runs beginning with the final four rules from the start symbol S.

It remains to verify that the halting protocol cannot be generated by runs beginning with the first rule $S \Rightarrow aYT$. However, any such run generates only words not beginning with $\#$ and words with a prefix from (I); by definition, no such word is the halting protocol. ◀

We can now define an appropriate *reducing function*:

$$f\colon (\mathtt{M}, x) \mapsto G_{\mathtt{M},x} \tag{2}$$

Of course, this function is total and computable. Moreover it allows us to deduce effective inseparability of $\mathbf{R}$ and $\bar{\mathbf{T}}$ from that of $\mathbf{C}$ and $\mathbf{H}$:

▶ **Proposition 22.** *The function f in (2) is an e.i.-reduction from $(\mathbf{R}, \bar{\mathbf{T}})$ to $(\mathbf{C}, \mathbf{H})$.*

Proof. Clearly $\mathbf{R} \cap \bar{\mathbf{T}} = \varnothing$. So we need to show the following two properties:

- if $(\mathtt{M}, x) \in \mathbf{C}$, then $f((\mathtt{M}, x)) = G_{\mathtt{M},x} \in \mathbf{R}$;
- if $(\mathtt{M}, x) \in \mathbf{H}$, then $f((\mathtt{M}, x)) = G_{\mathtt{M},x} \in \bar{\mathbf{T}}$.

The second property follows immediately from Proposition 21, since the halting protocol cannot be generated by $G_{\mathtt{M},x}$. The first property is more involved.

Let $\mathtt{M}$ loop on x and let $n+1$ be the length of an (incomplete) protocol of $\mathtt{M}$ on x which reaches a configuration where $\mathtt{M}$ is in state q_c. We shall show that $G_{\mathtt{M},x}$ is regularly total, according to the given n. That is, we will show that:

1. S generates all nonempty words of length $\leq n+1$.
2. Y generates all words of length n.

For (1), since $\mathtt{M}$ does not halt on x, there is no halting protocol, and S generates all nonempty words, by Proposition 21. In particular, S generates all words of length $\leq n+1$.

For (2) we must show that any word $\vec{a}$ of length n is in (I)′, i.e. that $\#\vec{a} \in$(I). For this, even if $\#\vec{a}$ is a prefix of the (infinite) protocol of $\mathtt{M}$ on x, it includes, by choice of n, the letter q_c, so $\#\vec{a} \in$(I). ◀

Now we can finally conclude the main result of this section:

Proof of Theorem 19. Follows immediately from Propositions 18, 20, and 22. ◀

4 Undecidability of (extensions of) $\mu\mathsf{S} + \mathsf{idem}$

This section is devoted to a proof of our first main result:

▶ **Theorem 23.** *Any r.e. extension of $\mu\mathsf{S} + \mathsf{idem}$ that is sound for $\mathcal{L}$ is Σ^0_1-complete.*

We shall actually strengthen this result in the next section to avoid the need for idempotency (and even identity), by a sort of reduction, but we present this weaker version to illustrate the main ideas in a simpler setting.

To prove Theorem 23, we first some intermediate results, reasoning about (productive) CFGs in $\mu\mathsf{S}(+\mathsf{idem})$. As we shall later consider settings without idempotency and identity, we shall be careful to avoid relying on the multiplicative identity 1, and shall also be sensitive to the use of idempotency.

▶ **Lemma 24.** *Fix a CFG G and write e_X for (any) corresponding solutions in $\mu\mathsf{S}$ of each non-terminal X of G, cf. Corollary 8. If X generates a word $\vec{a}$ then $\mu\mathsf{S} + \mathsf{idem} \vdash \vec{a} \leq e_X$.*

Proof. By induction on the derivation tree of $\vec{a}$ from X in G. Working in $\mu\mathsf{S} + \mathsf{idem}$:

- If $X \Rightarrow \vec{a}$, then we have $\vec{a} + g = e_X$ for some g by Corollary 8, and so $\vec{a} \leq e_X$ by idempotency of $\vec{a}$ and Proposition 9.
- If $X \Rightarrow \vec{a}_0 X_1 \vec{a}_1 \cdots \vec{a}_{n-1} X_n \vec{a}_n$, s.t. $\vec{a} = \vec{a}_0 \vec{b}_1 \vec{a}_1 \cdots \vec{b}_n \vec{a}_n$, then each $\vec{b}_i$ is generated by X_i by smaller derivation trees. Thus, by inductive hypothesis we have $\vec{b}_i \leq e_{X_i}$ for all i. Finally by monotonicity of products wrt $\leq$, we have indeed $\vec{a} \leq e_X$. ◀

Let us write $\top := \mu X \left(\sum_{a \in \mathcal{A}} a + \sum_{a \in \mathcal{A}} aX \right)$, which trivially computes the "total" language of all nonempty words.

▶ **Lemma 25.** *For $n > 0$, we have $\mu\mathsf{S} \vdash \top = \sum_{1 \leq |\vec{a}| \leq n} \vec{a} + \sum_{|\vec{a}|=n} \vec{a}\top$.*

Proof. By induction on $n > 0$:

- When $n = 1$, the statement follows by just an unfolding of the fixed point $\top$, axiom (ii).
- For the inductive step, we have:

$$
\begin{aligned}
\top &= \sum_{1 \leq |\vec{a}| \leq n} \vec{a} + \sum_{|\vec{a}|=n} \vec{a}\top && \text{by inductive hypothesis} \\
&= \sum_{1 \leq |\vec{a}| \leq n} \vec{a} + \sum_{|\vec{a}|=n} \vec{a} \left(\sum_{a \in \mathcal{A}} a + \sum_{a \in \mathcal{A}} a\top \right) && \text{by unfolding } \top\text{, axiom (ii)} \\
&= \sum_{1 \leq |\vec{a}| \leq n} \vec{a} + \sum_{|\vec{a}|=n} \vec{a} \sum_{a \in \mathcal{A}} a + \sum_{|\vec{a}|=n} \vec{a} \sum_{a \in \mathcal{A}} a\top && \text{by distributivity} \\
&= \sum_{1 \leq |\vec{a}| \leq n} \vec{a} + \sum_{|\vec{a}|=n+1} \vec{a} + \sum_{|\vec{a}|=n+1} \vec{a}\top && \text{by distributivity} \\
&= \sum_{1 \leq |\vec{a}| \leq n+1} \vec{a} + \sum_{|\vec{a}|=n+1} \vec{a}\top && \text{by rebracketing sums}
\end{aligned}
$$

◀

Note that $\top = \mu X \left(\sum_{a \in \mathcal{A}} a + \sum_{a \in \mathcal{A}} aX \right)$ is a solution of the non-terminal T in any regularly total grammar. In what follows, whenever we fix solutions of a regularly total grammar, we assume the solution for T is $\top$. Again, we shall be sensitive to the use of idempotency, making clear any such instances for our strengthening in the next section.

▶ **Proposition 26.** *Fix a productive CFG $G \in \mathbf{R}$ and let e_X be solutions of each of its non-terminals X in $\mu\mathsf{S}$, cf. Corollary 8, with $e_T = \top$. Then $\mu\mathsf{S} + \mathsf{idem} \vdash \top \leq e_S$.*

Proof. Since $G \in \mathbf{R}$, let $n > 0$ be such that:

1. S generates all nonempty words of length $\leq n+1$.
2. Y generates all words of length n.
3. G has a production $S \to aYT$ for each $a \in \mathcal{A}$.

We work in $\mu\mathsf{S} + \mathsf{idem}$. By 2 and Lemma 24 we have $\vec{a} \leq e_Y$ if $|\vec{a}| = n$. By idempotency of e_Y and Proposition 9 we have $\sum_{|\vec{a}|=n} \vec{a} \leq e_Y$. Thus by monotonicity of $\cdot$ and distributivity:

$$\sum_{|\vec{a}|=n} \vec{a}\top \leq e_Y\top$$

Now by (3) and Corollary 8 we have $g + \sum_{a\in\mathcal{A}} ae_Y\top = e_S$ for some g (and since e_T is $\top$). So by idempotency of $\sum_{a\in\mathcal{A}} ae_Y\top$ we have $\sum_{a\in\mathcal{A}} ae_Y\top \leq e_S$. Putting the two displays above together we have $\sum_{a\in\mathcal{A}} a \sum_{|\vec{a}|=n} \vec{a}\top \leq e_S$ by monotonicity, and so by distributivity:

$$\sum_{|\vec{a}|=n+1} \vec{a}\top \leq e_S \tag{3}$$

Now by (1) and again by Lemma 24 we have $\vec{a} \leq e_S$ whenever $1 \leq |\vec{a}| \leq n+1$. Again by idempotency of e_S and monotonicity of $+$ we have $\sum_{|\vec{a}|=1}^{n+1} \vec{a} \leq e_S$. Putting this together with (3) above, we have by idempotency of e_S and Proposition 9:

$$\sum_{|\vec{a}|=1}^{n+1} \vec{a} + \sum_{|\vec{a}|=n+1} \vec{a}\top \leq e_S$$

From here we conclude by Lemma 25 and transitivity of $\leq$, Proposition 9. ◀

We are now ready to prove the main result of this section:

Proof of Theorem 23. Let us assume that all CFGs have a start symbol S and write e_X^G for the solution of the nonterminal X in G obtained by Corollary 8. Let $\mathsf{U} \supseteq \mu\mathsf{S} + \mathsf{idem}$ be r.e. with $\mathcal{L} \models \mathsf{U}$. Writing $\mathsf{Tot}(\mathsf{U}) := \{G \text{ productive} : \mathsf{U} \vdash \top \leq e_S^G\}$, we have:

- $\mathsf{Tot}(\mathsf{U}) \cap \bar{\mathbf{T}} = \varnothing$, by assumption that $\mathcal{L} \models \mathsf{U}$.
- $\mathbf{R} \subseteq \mathsf{Tot}(\mathsf{U})$, since $\mu\mathsf{S} + \mathsf{idem} \subseteq \mathsf{U}$ and by Proposition 26.

Thus by Theorems 17 and 19 we have that $\mathsf{Tot}(\mathsf{U})$ is Σ_1^0-complete. Since there is an obvious computable reduction from $\mathsf{Tot}(\mathsf{U})$ to U, we have that U is Σ_1^0-complete too, as required. ◀

Since all theories we have introduced are r.e. and modelled by $\mathcal{L}$, we immediately have:

▶ **Corollary 27.** *The equational theories of idempotent Conway μ-semirings, idempotent Park μ-semirings, and Chomsky algebras are undecidable (in fact, Σ_1^0-complete).*

5 The case without idempotency

In this section, we shall strengthen the main result of the previous section to cover theories of semirings that are not necessarily idempotent.

5.1 Identity-free μ-semirings and productive expressions

The axiomatisation $\mu\mathsf{S}_{\not 1}$ is defined just like $\mu\mathsf{S}$ but without any axioms mentioning 1 (and without 1 in the language). The **productive** expressions, written p, q etc., are generated by,

$$p, q, \ldots \quad ::= \quad 0 \quad | \quad a \quad | \quad p + q \quad | \quad pe \quad | \quad ep \quad | \quad \mu X p \tag{4}$$

where e ranges over (possibly open, not necessarily productive) 1-free expressions.

By inspection of the proof of Proposition 7, we can give a refinement of Corollary 8:

▶ **Proposition 28.** *A productive CFG has productive solutions in $\mu\mathsf{S}_{\not 1}$. I.e., given a productive CFG $\{X_i \Rightarrow f_{ij}(\vec{X})\}_{i<n,j}$, there are productive expressions $\vec{e} = e_0, \ldots, e_{n-1}$ s.t. $\mu\mathsf{S}_{\not 1} \vdash e_i = \sum_j f_{ij}(\vec{e})$ for all $i < n$.*

In particular, note that the RHS of a production in a productive CFG is a productive expression, whence the same proof of Proposition 7 applies for the statement above. Note also that the proof does not rely on any properties of the multiplicative identity 1 (which in any case plays no role in productive CFGs).

5.2 An idempotent interpretation of productive expressions

In this section, we will define for every expression an expression which has the same language interpretation but is provably idempotent in $\mu\mathsf{S}_{\not 1}$. Temporarily write $e' := \mu X(e + \mu Y(X + Y))$. It is easy to see that, over $\mu\mathsf{S}_{\not 1}$, e' is idempotent:

$$\begin{aligned}
e' &= e + \mu Y(e' + Y) && \text{by unfolding } e', \text{ by axiom (ii)} \\
&= e + e' + \mu Y(e' + Y) && \text{by unfolding } \mu Y(e' + Y), \text{ by axiom (ii)} \\
&= (e + \mu Y(e' + Y)) + e' && \text{by associativity and commutativity of } + \\
&= e' + e' && \text{by the first line}
\end{aligned}$$

In the presence of an order e' is also an upper bound of e, and in the presence of induction, e' is in fact the *least* idempotent upper bound of e. In particular, in $\mathcal{L}$ (and any other idempotent model) we have simply $e' = e$:

- On one hand, we have $\mathcal{L}(e') = \mathcal{L}(e + \mu Y(e' + Y))$, and so $\mathcal{L}(e') \supseteq \mathcal{L}(e)$.
- On the other hand, we have that e solves the matrix of e':
  - Clearly $\mathcal{L}(\mu Y(e + Y)) = \mathcal{L}(e)$: as above we have $\mathcal{L}(\mu Y(e + Y)) = \mathcal{L}(e + \mu Y(e + Y)) \supseteq \mathcal{L}(e)$; we also have that e solves the matrix of $\mu Y(e + Y)$ in $\mathcal{L}$, i.e., $\mathcal{L}(e) = \mathcal{L}(e + e)$, thanks to idempotency.
  - Thus we have $e = e + e = e + \mu Y(e + Y)$ in $\mathcal{L}$, again by idempotency.

  Thus indeed $\mathcal{L}(e') \subseteq \mathcal{L}(e)$, as e' must be a least solution.

▶ **Definition 29** (Idempotent interpretation). *For 1-free expressions e, define e^i as follows:*

$$\begin{aligned}
0^i &:= 0 & (e + f)^i &:= e^i + f^i \\
a^i &:= \mu X(a + \mu Y(X + Y)) & (ef)^i &:= e^i f^i \\
X^i &:= X & (\mu X e)^i &:= \mu X e^i
\end{aligned}$$

As previously argued, it is not hard to see that:

▶ **Proposition 30.** $\mathcal{L}(e^i) = \mathcal{L}(e)$.

Proof sketch. By induction on the structure of e. In fact, we must strengthen the statement to $\mathcal{L}(e^i(\vec{A})) = \mathcal{L}(e(\vec{A}))$ for all languages $\vec{A}$. The case when e is some $a \in \mathcal{A}$ follows from the argument above. Every other case is immediate, with the μ-case following by the strong μ-congruence principle in $\mathcal{L}$: $\forall X\, e(X) = f(X) \to \mu X e(X) = \mu X f(X)$. ◀

Moreover, $\cdot^i$ induces a bona fide interpretation of identity-free μ-semirings:

▶ **Proposition 31.** $\mu\mathsf{S}_\chi \vdash e = f \implies \mu\mathsf{S}_\chi \vdash e^i = f^i$.

To see this, simply replace every expression e in a $\mu\mathsf{S}_\chi$ proof by e^i. It remains to verify that the axioms and rules remain correct, which is routine. Finally, thanks to the fact that idempotent elements of a semiring form an ideal, cf. Fact 11, we have idempotency of the image of $\cdot^i$ on productive expressions provably in $\mu\mathsf{S}_\chi$:

▶ **Lemma 32.** *If $e(\vec{X})$ is productive then $\mu\mathsf{S}_\chi \vdash e^i(\vec{f}) + e^i(\vec{f}) = e^i(\vec{f})$ for all $\vec{f}$.*

Proof. By induction on the structure of $e(\vec{X})$, according to the grammar of productive expressions in (4). Working in $\mu\mathsf{S}_\chi$ we have:

- If $e(\vec{X})$ is 0 then:

$$\begin{aligned} 0^i + 0^i &= 0 + 0^i && \text{by definition of } 0^i \\ &= 0^i && \text{since 0 is an additive unit} \end{aligned}$$

- If $e(\vec{X})$ is some $a \in \mathcal{A}$ then $a^i + a^i = a^i$ by the argument at the beginning of this subsection.
- If $e(\vec{X})$ is $p(\vec{X}) + q(\vec{X})$ then:

$$\begin{aligned} & (p^i(\vec{f}) + q^i(\vec{f})) + (p^i(\vec{f}) + q^i(\vec{f})) \\ = \;& (p^i(\vec{f}) + p^i(\vec{f})) + (q^i(\vec{f}) + q^i(\vec{f})) && \text{by associativity and commutativity of } + \\ = \;& p^i(\vec{f}) + q^i(\vec{f}) && \text{by inductive hypothesis for } p(\vec{x}) \text{ and } q(\vec{x}) \end{aligned}$$

- If $e(\vec{X})$ is $p(\vec{X})g(\vec{X})$ then:

$$\begin{aligned} & p^i(\vec{f})g^i(\vec{f}) + p^i(\vec{f})g^i(\vec{f}) \\ = \;& (p^i(\vec{f}) + p^i(\vec{f}))g^i(\vec{f}) && \text{by distributivity} \\ = \;& p^i(\vec{f})g^i(\vec{f}) && \text{by inductive hypothesis for } p(\vec{x}) \end{aligned}$$

- If $e(\vec{X})$ is $g(\vec{X})p(\vec{X})$ then the argument is symmetric to above.
- If $e(\vec{X})$ is $\mu X p(X, \vec{X})$ then:

$$\begin{aligned} & \mu X p^i(X, \vec{f}) + \mu X p^i(X, \vec{f}) \\ = \;& p^i(\mu X p^i(X, \vec{f}), \vec{f}) + p^i(\mu X p^i(X, \vec{f}), \vec{f}) && \text{by } \mu\text{-unfolding, axiom (ii)} \\ = \;& p^i(\mu X p^i(X, \vec{f}), \vec{f}) && \text{by inductive hypothesis for } p(X, \vec{X}) \\ = \;& \mu X p^i(X, \vec{f}) && \text{by } \mu\text{-unfolding, axiom (ii)} \end{aligned}$$

◀

Note that, the above result notwithstanding, e^i is typically not productive, even if e is. In particular, a^i is not productive, even though a is.

Notice that, Proposition 31 notwithstanding, $\cdot^i$ does not necessarily interpret $\mu\mathsf{S}_\chi + \mathsf{idem}$ into $\mu\mathsf{S}_\chi$. This is because, a priori, a proof in $\mu\mathsf{S}_\chi + \mathsf{idem}$ may exploit idempotency of expressions not in the image of $\cdot^i$ on productive expressions. For this reason, in what follows, we shall exploit the fact that we carefully kept track of the use of idempotency in our arguments in the previous section.

5.3 Undecidability of $\mu\mathsf{S}_\chi$ (and extensions)

Recall $\top := \mu X \left(\sum \mathcal{A} + \sum_{a \in \mathcal{A}} aX \right)$. By a similar argument to Proposition 26 in the previous section, we can obtain:

▶ **Proposition 33.** *Let $G \in \mathbf{R}$ and let e_X be productive solutions of each of its non-terminals X in $\mu\mathsf{S}_\chi$, cf. Proposition 28, with $e_T = \top$. Then $\mu\mathsf{S}_\chi \vdash \top^i \leq e_S^i$.*

For this, we need appropriate versions of the two intermediate results we employed, Lemmas 24 and 25, in the previous section. Their proofs are morally similar to the previous ones, but we must be careful to state them appropriately in the absence of idempotency, exploiting the refined results of this section on productive CFGs and $\mu\mathsf{S}_\chi$

▶ **Lemma 34.** *Fix a productive CFG G and write e_X for (any) corresponding productive solutions in $\mu\mathsf{S}_\chi$ of each non-terminal X of G, cf. Proposition 28. If $X \to^* \vec{a}$ in G then $\mu\mathsf{S}_\chi \vdash \vec{a}^i \leq e_X^i$.*

Proof. By induction on the derivation tree of $\vec{a}$ from X in G. Working in $\mu\mathsf{S}_\chi$:

- If $X \to \vec{a}$ then we have $\vec{a} + g = e_X$ for some g by Proposition 28. Thus $\vec{a}^i + g^i = e_X^i$ by Proposition 31, and so $\vec{a}^i \leq e_X^i$ by idempotency of $\vec{a}^i$, Lemma 32.
- The inductive step is the same as in the proof of Lemma 24. ◀

▶ **Lemma 35.** *For $n > 0$, we have $\mu\mathsf{S}_\chi \vdash \top^i = \sum_{|\vec{a}|=1}^{n} \vec{a}^i + \sum_{|\vec{a}|=n} \vec{a}^i \top^i$.*

Proof. The proof of Lemma 25 does not use any identity axioms, and so goes through already in $\mu\mathsf{S}_\chi$. The result now follows by Proposition 31. ◀

We are now ready to prove the result we stated at the beginning of this section:

Proof of Proposition 33. The proof goes through just like the proof of Proposition 26, only replacing all (productive) expressions by their image under $\cdot^i$. All appeals to Lemmas 24 and 25 are replaced by Lemmas 34 and 35 respectively. Note that, by inspection of the proof, the only uses of idempotency (marked in purple) are on productive expressions, whose image under $\cdot^i$ is indeed provably idempotent in $\mu\mathsf{S}_\chi$ by Lemma 32. In particular, the expressions e_Y and e_S are productive by assumption. ◀

5.4 Putting it all together

Finally, we arrive at our promised strengthening of Theorem 23:

▶ **Theorem 36.** *Any r.e. extension of $\mu\mathsf{S}_\chi$ sound for $\mathcal{L}$ is Σ_1^0-complete.*

The proof is similar to that of Theorem 23, only more careful in handling the $\cdot^i$ translation.

Proof. Like in the proof of Theorem 23, let us assume that all CFGs have a start symbol S and write e_X^G for the productive solutions of nonterminals X in a productive grammar G obtained by Proposition 28. Let $\mathsf{U} \supseteq \mu\mathsf{S}_\chi$ be r.e. with $\mathcal{L} \models \mathsf{U}$. Now we write $\mathsf{Tot}(\mathsf{U}) := \{G \text{ productive} : \mathsf{U} \vdash \top^i \leq (e_S^G)^i\}$. We have:

- $\mathsf{Tot}(\mathsf{U}) \cap \bar{\mathbf{T}} = \varnothing$, by assumption that $\mathcal{L} \models \mathsf{U}$ and by Proposition 30.
- $\mathbf{R} \subseteq \mathsf{Tot}(\mathsf{U})$, since $\mu\mathsf{S}_\chi \subseteq \mathsf{U}$ and by Proposition 33.

Thus by Theorems 17 and 19 we have that $\mathsf{Tot}(\mathsf{U})$ is Σ_1^0-complete. Again, by the obvious computable reduction from $\mathsf{Tot}(\mathsf{U})$ to U, we conclude as required. ◀

▶ **Corollary 37.** *The equational theories of Conway μ-semirings and Park μ-semirings are undecidable (and, in fact, Σ_1^0-complete). The same holds even for their identity-free versions.*

6 Conclusions

In this work, we addressed the question of (un)decidability of theories of various classes of semirings with fixed points. Using recursion-theoretic methods, we showed that a large class of such theories are undecidable. These results account for many of the μ-semiring theories proposed in previous works, cf. [20, 6, 7, 8].

Conway semirings were proposed in [6, 7] as a basic finite equational theory that is nonetheless powerful enough to carry out reduction to Greibach normal form. Our basic system $\mu\mathsf{S}$ is weaker/more general than Conway semirings, and is also finitely equationally axiomatised. It would be interesting to examine if it may have any further utility, other than as a lower bound for the range of our undecidability results herein. For instance, are there results of formal language theory that are validated already in $\mu\mathsf{S}$?

Finally, our results do not cover pure μ-semirings, where $\mu Xe(X)$ is not necessarily a fixed point. The (un)decidability of its equational theory is, as far as we know, open.

References

1 Hans Bekić. Definable operations in general algebras, and the theory of automata and flowcharts. In C. B. Jones, editor, *Programming Languages and Their Definition: H. Bekić (1936–1982)*, pages 30–55. Springer, Berlin, Heidelberg, 1984. doi:10.1007/BFb0048939.

2 Bruno Courcelle. Equivalences and transformations of regular systems—Applications to recursive program schemes and grammars. *Theoretical Computer Science*, 42:1–122, 1986. doi:10.1016/0304-3975(86)90050-2.

3 Anupam Das and Abhishek De. A proof theory of (ω-)context-free languages, via non-wellfounded proofs. In Christoph Benzmüller, Marijn J.H. Heule, and Renate A. Schmidt, editors, *Automated Reasoning, IJCAR 2024*, volume 14740 of *Lectures Notes in Computer Science*, pages 237–256, Cham, 2024. Springer. doi:10.1007/978-3-031-63501-4_13.

4 Amina Doumane, Denis Kuperberg, Damien Pous, and Cécilia Pradic. Kleene algebra with hypotheses. In Mikołaj Bojańczyk and Alex Simpson, editors, *Foundations of Software Science and Computation Structures, FoSSaCS 2019*, volume 11425 of *Lecture Notes in Computer Science*, pages 207–223, Cham, 2019. Springer. doi:10.1007/978-3-030-17127-8_12.

5 Zoltán Ésik and Werner Kuich. Modern automata theory, 2012. doi:10.34726/2481.

6 Zoltán Ésik and Hans Leiß. Greibach normal form in algebraically complete semirings. In Julian Bradfield, editor, *Computer Science Logic, CSL 2002*, volume 2471 of *Lecture Notes in Computer Science*, pages 135–150, Berlin, Heidelberg, 2002. Springer. doi:10.1007/3-540-45793-3_10.

7 Zoltán Ésik and Hans Leiß. Algebraically complete semirings and Greibach normal form. *Annals of Pure and Applied Logic*, 133(1):173–203, 2005. Festschrift on the occasion of Helmut Schwichtenberg's 60th birthday. doi:10.1016/j.apal.2004.10.008.

8 Niels Bjørn Bugge Grathwohl, Fritz Henglein, and Dexter Kozen. Infinitary axiomatization of the equational theory of context-free languages. *Fundamenta Informaticae*, 150(3–4):241–257, 2013. doi:10.3233/FI-2017-1469.

9 Sheila A. Greibach. A new normal-form theorem for context-free phrase structure grammars. *Journal of the ACM*, 12(1):42–52, 1965. doi:10.1145/321250.321254.

10 Jozef Gruska. A characterization of context-free languages. *Journal of Computer and System Sciences*, 5(4):353–364, 1971. doi:10.1016/S0022-0000(71)80023-5.

11 John E. Hopcroft and Jeff D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Publishing Company, 1979.

12 Mark Hopkins. The algebraic approach I: The algebraization of the Chomsky hierarchy. In Rudolf Berghammer, Bernhard Möller, and Georg Struth, editors, *Relations and Kleene Algebra in Computer Science, RelMiCS 2008*, volume 4988 of *Lecture Notes in Computer Science*, pages 155–172, Berlin, Heidelberg, 2008. Springer. doi:10.1007/978-3-540-78913-0_13.

13 Mark Hopkins. The algebraic approach II: Dioids, quantales and monads. In Rudolf Berghammer, Bernhard Möller, and Georg Struth, editors, *Relations and Kleene Algebra in Computer Science, RelMiCS 2008*, volume 4988 of *Lecture Notes in Computer Science*, pages 173–190, Berlin, Heidelberg, 2008. Springer. doi:10.1007/978-3-540-78913-0_14.

14 Dexter Kozen. A completeness theorem for Kleene algebras and the algebra of regular events. *Information and Computation*, 110(2):366–390, 1994. doi:10.1006/inco.1994.1037.

15 Dexter Kozen. *Automata and Complexity*. Springer-Verlag, New York, 1997.

16 Dexter Kozen. On the complexity of reasoning in Kleene algebra. *Information and Computation*, 179(2):152–162, 2002. doi:10.1006/inco.2001.2960.

17 Stepan Kuznetsov. The logic of action lattices is undecidable. In *Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '19. IEEE Press, 2019. doi:10.1109/LICS.2019.8785659.

18 Stepan Kuznetsov. Action logic is undecidable. *ACM Transactions on Computational Logic*, 22(2):10, 26 pp., 2021. doi:10.1145/3445810.

19 Stepan L. Kuznetsov. Commutative action logic. *Journal of Logic and Computation*, 33(6):1427–1462, 2023. doi:10.1093/logcom/exac008.

20 Hans Leiß. Towards Kleene Algebra with recursion. In Egon Börger, Gerhard Jäger, Hans Kleine Büning, and Michael M. Richter, editors, *Computer Science Logic, CSL 1991*, volume 626 of *Lecture Notes in Computer Science*, pages 242–256, Berlin, Heidelberg, 1992. Springer. doi:10.1007/BFb0023771.

21 Hans Leiss. The matrix ring of a μ-continuous Chomsky algebra is μ-continuous. In Jean-Marc Talbot and Laurent Regnier, editors, *25th EACSL Annual Conference on Computer Science Logic (CSL 2016)*, volume 62 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:15, Dagstuhl, Germany, 2016. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CSL.2016.6.

22 John Myhill. Creative sets. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 1:97–108, 1955.

23 Hartley Rogers. *Theory of Recursive Functions and Effective Computability*. MIT Press, 1987.

24 Michael Sipser. *Introduction to the Theory of Computation*. International Thomson Publishing, 1st edition, 1996.

A Decidability for semirings (without μ)

Write S for the theory of semirings over μ-free expressions. In fact S admits a *free model* by finite multisets of words. This is essentially a weighted version of the language model $\mathcal{L}$.

▶ **Definition 38** (Multiset model). *Write $\mathcal{M}$ for the set of finite multisets over A^*. For each μ-free closed expression e we define the finite multiset $\mathcal{M}(e)$ over A^* as follows:*

$$\begin{array}{llll} \mathcal{M}(0) := \varnothing & \mathcal{M}(a) := \{a\} & & \\ \mathcal{M}(1) := \{\varepsilon\} & \mathcal{M}(e+f) := \mathcal{M}(e) \cup \mathcal{M}(f) & \mathcal{M}(ef) & = \{\vec{a}\vec{b} : \vec{a} \in \mathcal{M}(e), \vec{b} \in \mathcal{M}(f)\} \end{array}$$

As expected, $\mathcal{M}$ indeed forms a semiring:

▶ **Proposition 39** (Soundness). $\mathcal{M} \models \mathsf{S}$.

Proof. Associativity and commutativity of $+$ reduces to the same for multiset union. $e+0 = e$ holds since $\varnothing$ is a unit for multiset union. $e1 = e = 1e$ holds as ε is a unit for word concatenation. $e0 = 0 = 0e$ holds by unfolding the definition. Finally the distributivity laws are also readily verified by unfolding definitions, reducing to distributivity of logical operations on multisets. ◀

Via $\mathcal{M}$ we inherit a *canonical* representation of each expression. For each μ-free closed expression e define its **canonical polynomial** $p_e := \sum_{\vec{a} \in \mathcal{M}(e)} \prod \vec{a}$. It is straightforward to see that expressions are equivalent to their canonical polynomials in all semirings:

▶ **Lemma 40** (Canonical equivalence). $\mathsf{S} \vdash e = p_e$

Proof. By induction on e, reasoning in S:

- $p_0 = \sum_{\vec{a} \in \varnothing} \prod \vec{a} = 0$
- $p_1 = \sum_{\vec{a} \in \{\varepsilon\}} \prod \vec{a} = \prod \varepsilon = 1$.
- $p_a = \sum_{\vec{a} \in \{a\}} \prod \vec{a} = \prod a = a$
- If $e = f + g$ we have:

$$\begin{aligned} p_e &= \textstyle\sum_{\vec{a} \in \mathcal{M}(f+g)} \prod \vec{a} && \text{by definition} \\ &= \textstyle\sum_{\vec{a} \in \mathcal{M}(f)} \prod \vec{a} + \sum_{\vec{a} \in \mathcal{M}(g)} \prod \vec{a} && \text{since } \mathcal{M}(f+g) = \mathcal{M}(f) \cup \mathcal{M}(g) \\ &= p_f + p_g && \text{by definition} \\ &= f + g && \text{by inductive hypotheses} \end{aligned}$$

- If $e = fg$ we have:

$$\begin{aligned} p_e &= \textstyle\sum_{\vec{a} \in \mathcal{M}(fg)} \prod \vec{a} && \text{by definition} \\ &= \textstyle\sum_{\vec{b} \in \mathcal{M}(f)} \sum_{\vec{c} \in \mathcal{M}(g)} \prod \vec{b}\vec{c} && \text{since } \mathcal{M}(fg) = \{\vec{b}\vec{c} : \vec{b} \in \mathcal{M}(f), \vec{c} \in \mathcal{M}(g)\} \\ &= \textstyle\sum_{\vec{b} \in \mathcal{M}(f)} \prod \vec{b} \cdot \sum_{\vec{c} \in \mathcal{M}(g)} \prod \vec{c} && \text{by distributivity laws of } \mathsf{S} \\ &= p_f p_g && \text{by definition} \\ &= fg && \text{by inductive hypotheses} \end{aligned}$$

◀

▶ **Theorem 41** (Free model). $\mathsf{S} \vdash e = f \iff \mathcal{M}(e) = \mathcal{M}(f)$.

Proof. The $\implies$ direction is already immediate from Proposition 39. For the $\impliedby$ direction, if $\mathcal{M}(e) = \mathcal{M}(f)$ then syntactically we have $p_e = p_f$,[4] whence $\mathsf{S} \vdash e = f$ by Lemma 40. ◀

▶ **Corollary 42.** S *is decidable.*

Proof. By Theorem 41 we can check $\mathsf{S} \vdash_? e = f$ by comparing $\mathcal{M}(e), \mathcal{M}(f)$ for equality. ◀

[4] Implicitly we are working modulo associativity and commutativity of $+$ here.