On Polynomial-Time Decidability of k-Negations Fragments of FO Theories (Extended Abstract)

Christoph Haase
University of Oxford, UK

Alessio Mansutti
IMDEA Software Institute, Madrid, Spain

Amaury Pouly
University of Oxford, UK
Université Paris Cité, CNRS, IRIF, France

Abstract

This paper introduces a generic framework that provides sufficient conditions for guaranteeing polynomial-time decidability of fixed-negation fragments of first-order theories that adhere to certain fixed-parameter tractability requirements. It enables deciding sentences of such theories with arbitrary existential quantification, conjunction and a fixed number of negation symbols in polynomial time. It was recently shown by Nguyen and Pak [*SIAM J. Comput.* 51(2): 1–31 (2022)] that an even more restricted such fragment of Presburger arithmetic (the first-order theory of the integers with addition and order) is NP-hard. In contrast, by application of our framework, we show that the fixed negation fragment of weak Presburger arithmetic, which drops the order relation from Presburger arithmetic in favour of equality, is decidable in polynomial time.

2012 ACM Subject Classification Theory of computation

Keywords and phrases first-order theories, arithmetic theories, fixed-parameter tractability

Digital Object Identifier 10.4230/LIPIcs.MFCS.2023.52

Funding This work is part of a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 852769, ARiAT).

1 Introduction

It is well-known that even the simplest first-order theories are computationally difficult to decide [12]. In particular, it follows from a result of Stockmeyer that every theory with a non-trivial predicate such as equality is PSPACE-hard to decide [24]. Even when restricting to existential fragments or fragments with a fixed number of quantifier alternations, deciding such fragments is NP-hard at best. There are two further kinds of restrictions that may lead to tractability. First, restricting the Boolean structure of the matrix of formulae in prenex form yields tractable fragments of, e.g., the Boolean satisfiability problem. For instance, the Horn and XOR-fragments of propositional logic are decidable in polynomial time, and this even applies to quantified Boolean Horn formulae, see e.g. [7]. Second, restricting the number of variables can also lead to tractable fragments of a first-order theory, especially for structures over infinite domains such as Presburger arithmetic, the first-order theory of the structure $(\mathbb{Z}, 0, 1, +, \leq)$. While the existential fragment of Presburger arithmetic is NP-complete in general [5,25], it becomes polynomial-time decidable when additionally fixing the number of variables [22]; this is a consequence of polynomial-time decidability of integer programming when the dimension is fixed [18]. Already when moving to an $\exists\forall$ quantifier prefix, Presburger arithmetic becomes NP-hard [23]. On the first sight, this result seems to preclude any possibility of further restrictions that may lead to tractable fragments of

Presburger arithmetic. However, another tractable fragment was identified in the context of investigating the complexity of the classical *Frobenius problem*. Given positive integers $a_1, \ldots, a_n \in \mathbb{N}$, this problem is to determine the largest integer that cannot be obtained as a non-negative linear combination of the a_i, which is called the *Frobenius number*. For any fixed $n > 0$, deciding whether the Frobenius number exceeds a given threshold can be reduced to the so-called *short fragment* of Presburger arithmetic, a highly restricted fragment in which everything, the number of atomic formulae and the number of variables (and *a fortiori* the number of quantifier alternations), is fixed – except for the coefficients of variables appearing in linear terms of atomic inequalities. Kannan [15] showed that the $\forall^k\exists^\ell$-fragment of short Presburger arithmetic is decidable in polynomial time for all fixed k, ℓ, which implies that the decision version of the Frobenius problem is in polynomial time for fixed n. However, in a recent breakthrough, Nguyen and Pak showed that there are fixed k, ℓ, m such that the $\exists^k\forall^\ell\exists^m$-fragment of short Presburger arithmetic is NP-hard, and adding further (fixed) quantifier alternations allows the logic to climb the polynomial hierarchy [20].

The main contribution of this paper is to develop an algorithmic framework that enables us to show that *fixed negation fragments* of certain first-order theories are decidable in polynomial time. Formulae in this fragment are generated by the following grammar, where Ψ are atomic formulae of the underlying first-order theory, and an arbitrary but a priori fixed number of negation symbols is allowed to occur:

$$\Phi ::= \exists x\, \Phi \mid \neg\Phi \mid \Phi \wedge \Phi \mid \Psi\,.$$

We give sufficient conditions for the fixed negation fragment of a first-order theory to be decidable in polynomial time. Observe that this fragment is more permissive than the "short fragments" of Kannan, as it allows for an unbounded number of quantified variables and an unbounded number of conjunctions. However, it implicitly fixes the number of quantifier alternations as well as the number of disjunctions.

As a main application of our framework, we show that, unlike full Presburger arithmetic, the fixed negation fragment of weak Presburger arithmetic (weak PA) is polynomial-time decidable. Weak PA is the strictly less expressive substructure $(\mathbb{Z}, 0, 1, +, =)$ (which is, in fact, the structure Presburger studied in his seminal article [21]). It was recently shown that weak PA has the same complexity as Presburger arithmetic [9]. Bodirsky et al. showed that the weak PA fragment of (an unbounded number of) existential linear Horn equations $\bigwedge_{i\in I}(A_i \cdot \mathbf{x} = b_i) \to (C_i \cdot \mathbf{x} = d_i)$ over $\mathbb{Z}$ can be decided in PTIME [4]. It follows from the generic results in this paper that the quantified versions of those formulae with the number of quantifier alternations and $|I|$ fixed is also polynomial-time decidable. This is the best possible such result, since one can show that, for I unbounded, the $\exists\forall$ fragment of linear Horn equations in 2 variables is NP-hard (a proof of this is given in Appendix A). Our framework not only allows for deciding satisfiability (and validity) of fixed negation formulae of Weak PA in PTIME, but also to compute a representation of the set of solutions of a given formula.

Our algorithmic framework is parametric on a concrete representation of the sets definable within the first-order theory $\mathcal{T}$ under consideration and only requires a sensible representation of solution sets for conjunctions of atomic formulae. From this representation, the framework guides us to the definition of a companion structure $\mathcal{R}$ for the theory $\mathcal{T}$ in which function symbols and relations in $\mathcal{R}$ are interpreted as reductions from parametrised complexity theory, e.g. UXP reductions, see e.g. [11, Chapter 15]. By requiring mild conditions on the types of reductions and parameters that the functions and relations in $\mathcal{R}$ must have, we are able to give a general theorem for the tractability of the fixed negation satisfiability

and entailment problems for $\mathcal{T}$. One technical issue we show how to overcome in a general way is how to treat negation, which is especially challenging when the initial representation provided to the framework is not closed under complementation (as it is the case in our treatment of weak PA). We resolve this issue by reducing it to the problem of completing a prelattice under relative complement, which we analyse computationally. Our main source of inspiration here is the notion of the so-called *difference normal form* of propositional logic, a rather unorthodox normal form introduced by Hausdorff [13, Ch. 1§5].

2 Preliminaries

We assume familiarity with basic concepts from first-order logic, set theory and abstract algebra. This section focuses on notation and simple definitions that might be non-standard to some readers.

Sets and functions. We write $\mathrm{seq}(A)$ for the set of all finite tuples over a set A, and denote by $()$ the empty tuple. This definition corresponds to the standard notion of Kleene star A^* of a set A. The discrepancy in notation is introduced to avoid writing $(\Sigma^*)^*$ for the domain of all tuples of finite words over an alphabet Σ, as in formal languages the Kleene star comes equipped with the axiom $(\Sigma^*)^* = \Sigma^*$. We denote this domain by $\mathrm{seq}(\Sigma^*)$.

We write $f :\subseteq X \to Y$ (resp. $f\colon X \to Y$) to denote a *partial (resp. total) function* from X to Y. The domain of f is denoted with $\mathrm{dom}(f)$. We write $\mathrm{id}_A\colon A \to A$ for the identity function on A. Given $f :\subseteq A \to B$, $g :\subseteq B \to C$ and $h :\subseteq D \to E$, we denote by $(g \circ f) :\subseteq A \to C$ and $(f \times h) :\subseteq A \times D \to B \times E$ the *composition* and the *Cartesian product* of functions.

Structures with indexed families of functions. We consider an expansion to the traditional definition of structure from universal algebra that accommodate for a potentially infinite number of functions. As usual, a *structure* $\mathcal{A} = (A, \sigma, I)$ consists of a *domain* A (a set), a *signature* σ, and an *interpretation function* I; however in our case the signature is a quadruple $\sigma = (\mathcal{F}, \mathcal{G}, \mathcal{R}, ar)$ containing not only a set of *function symbols* $\mathcal{F}$, a set of *relation symbols* $\mathcal{R}$, and the *arity function* $ar\colon \mathcal{F} \uplus \mathcal{R} \uplus \mathcal{G} \to \mathbb{N}$, but also a set of (indexed) *families of function symbols* $\mathcal{G}$. Each element of $\mathcal{G}$ is a pair (g, X) where g is a function symbol and X is a countable set of indices. The interpretation function I associates to every $f \in \mathcal{F}$ a map $f^{\mathcal{A}}\colon A^{ar(f)} \to A$, to every $(g, X) \in \mathcal{G}$ a map $g^{\mathcal{A}}\colon X \times A^{ar(g)} \to A$, and to every $R \in \mathcal{R}$ a relation $R^{\mathcal{A}} \subseteq A^{ar(R)}$ which we often see as a function $R^{\mathcal{A}}\colon A^{ar(R)} \to \{\top, \bot\}$.

The standard notions of *homomorphism, embedding and isomorphism of structures*, as well as the notions of *congruence for a structure* and *quotient structure* extend in a natural way to structures having families of functions. For instance, a *homomorphism* from $\mathcal{A} = (A, \sigma, I)$ into $\mathcal{B} = (B, \sigma, J)$ is a map $h\colon A \to B$ that *preserves* all functions, families of functions and relations; so in particular given $(g, X) \in \mathcal{G}$, the map h satisfies $g^{\mathcal{B}}(x, h(a_1), \ldots, h(a_{ar(g)})) = h(g^{\mathcal{A}}(x, a_1, \ldots, a_{ar(g)}))$ for every $x \in X$ and $a_1, \ldots, a_{ar(g)} \in A$.

We denote structures in calligraphic letters $\mathcal{A}, \mathcal{B}, \ldots$ and their domains in capital letters $A, B, \ldots$. When the arity function ar and the interpretation I are clear from the context, we write $(A, f_1^{\mathcal{A}}, ..., f_j^{\mathcal{A}}, (g_1^{\mathcal{A}}, X_1), ..., (g_\ell^{\mathcal{A}}, X_\ell), R_1^{\mathcal{A}}, ..., R_k^{\mathcal{A}})$, and often drop the superscript $\mathcal{A}$, for a structure $\mathcal{A} = (A, \sigma, I)$ with $\sigma = (\{f_1, ..., f_j\}, \{(g_1, X_1), ..., (g_\ell, X_\ell)\}, \{R_1, ..., R_k\}, ar)$. A structure $\mathcal{A} = (A, \sigma, I)$ is said to be *an algebra of sets* whenever A is a family of sets and the symbols in σ are taken from $\{\emptyset, \cup, \cap, \setminus, \subseteq, =\}$ and interpreted as the canonical operations on sets. Since $\mathcal{A}$ is a structure, the set A is closed under all set operations in σ.

▶ **Example 1.** To understand the notion of families of function, consider the structure $\mathcal{A} = (\mathbb{Z}, (\text{mul}, \mathbb{N}))$. Here, the family of functions $(\text{mul}, \mathbb{N})$, interpreted as $\text{mul}^{\mathcal{A}}(n, x) = n \cdot x$ for all $n \in \mathbb{N}$ and $x \in \mathbb{Z}$, uniformly define multiplication by a non-negative constant n.

First-order theories (finite tuples semantics). The first-order (FO) language of the signature $\sigma = (\mathcal{F}, \mathcal{G}, \mathcal{R}, ar)$ is the set of formulae $\Phi, \Psi, \dots$ built from the grammar

$$\Phi, \Psi \coloneqq r(t_1, \dots, t_{ar(r)}) \mid \neg\Phi \mid \Phi \wedge \Psi \mid \exists x.\Phi, \qquad t \coloneqq x \mid f(t_1, \dots, t_{ar(f)}) \mid g(i, t_1, \dots, t_{ar(g)}),$$

where $x \in \mathbb{V}$ is a first-order variable, $r \in \mathcal{R}$, $f \in \mathcal{F}$, $(g, X) \in \mathcal{G}$ and $i \in X$ (more precisely, i belongs to a representation of X, see Section 3). Lexemes of the form $r(t_1, \dots, t_{ar(r)})$ are the *atomic formulae* of the language. Throughout the paper, we implicitly assume an order on the variables in $\mathbb{V}$, and write x_j for the j-th variable (indexed from 1).

For our purposes it comes handy to define the FO theory of a structure $\mathcal{A} = (A, \sigma, I)$ using tuples instead of the more standard approach of having maps from variables to values. The two definitions are equivalent. Given an atomic formula $r(t_1, \dots, t_{ar(r)})$ having x_n as the largest appearing variable, we write $[\![r(t_1, ..., t_{ar(r)})]\!]_{\mathcal{A}} \subseteq A^n$ for the set of n-tuples, corresponding to values of the first n variables, that makes $r(t_1, ..., t_{ar(r)})$ true under the given interpretation I. Let $\mathbf{I} \coloneqq \{(i_1, ..., i_k) \in \text{seq}(\mathbb{N}) : i_1, ..., i_k \text{ all distinct}\}$. The *first-order theory* of $\mathcal{A}$ is the structure $\text{FO}(\mathcal{A}) \coloneqq ([\![\mathcal{A}]\!]_{\text{FO}}, \bot, \top, \vee, \wedge, -, (\pi, \mathbf{I}), (\pi^{\forall}, \mathbf{I}), \leq)$, where:

1. $[\![\mathcal{A}]\!]_{\text{FO}}$ is the least set that contains $[\![r(t_1, ..., t_{ar(r)})]\!]_{\mathcal{A}}$, for each formula $r(t_1, ..., t_{ar(r)})$, and that is closed under the functions $\bot, \top, \vee, \wedge, -, (\pi, \mathbf{I})$ and $(\pi^{\forall}, \mathbf{I})$, defined below;
2. $\bot$ is interpreted as $\emptyset$, $\top$ is interpreted as $\{()\}$, and given $S \subseteq A^n$ and $T \subseteq A^m$,

$$\begin{aligned}
&S \vee T \coloneqq \{(a_1, \dots, a_{\max(n,m)}) : (a_1, \dots, a_n) \in S \text{ or } (a_1, \dots, a_m) \in T\},\\
&S \wedge T \coloneqq \{(a_1, \dots, a_{\max(n,m)}) : (a_1, \dots, a_n) \in S \text{ and } (a_1, \dots, a_m) \in T\},\\
&S - T \coloneqq \{(a_1, \dots, a_{\max(n,m)}) : (a_1, \dots, a_n) \in S \text{ and } (a_1, \dots, a_m) \notin T\},\\
&\pi((i_1, ..., i_k), X) \coloneqq \{\gamma \in A^n : \text{there is } \mathbf{a} \in A^k \text{ such that } \gamma[(i_1, \dots, i_k) \leftarrow \mathbf{a}] \in X\},\\
&\pi^{\forall}((i_1, ..., i_k), X) \coloneqq \{\gamma \in A^n : \text{for every } \mathbf{a} \in A^k, \gamma[(i_1, \dots, i_k) \leftarrow \mathbf{a}] \in X\},\\
&S \leq T \text{ if and only if } S \times A^m \subseteq T \times A^n,
\end{aligned}$$

where $\gamma[(i_1, \dots, i_k) \leftarrow \mathbf{a}]$ is the tuple obtained from the n-tuple γ by replacing its i_j-th component with the j-th component of $\mathbf{a}$, for every $j \in [1, \min(k, n)]$.

The semantics $[\![.]\!]_{\mathcal{A}}$ of the first-order language of σ is extended to non-atomic formulae via $\text{FO}(\mathcal{A})$. As usual, for negation, conjunction and existential quantification, we have $[\![\neg\Phi]\!]_{\mathcal{A}} \coloneqq \top - [\![\Phi]\!]_{\mathcal{A}}$, $[\![\Phi \wedge \Psi]\!]_{\mathcal{A}} \coloneqq [\![\Phi]\!]_{\mathcal{A}} \wedge [\![\Phi]\!]_{\mathcal{A}}$, and $[\![\exists x_i.\Phi]\!]_{\mathcal{A}} \coloneqq \pi((i), [\![\Phi]\!]_{\mathcal{A}})$. We remark that $\text{FO}(\mathcal{A})$ contains operators whose syntactic counterpart is absent from the FO language of σ, such as the universal projection $\pi^{\forall}$. This is done for algorithmic purposes, as the framework we introduce in Section 4 treats these operators as first-class citizens.

We let $\text{AC}(\sigma)$ be the set of all conjunctions of atomic formulae in the FO language of σ.

Fixed negation fragments. Fix $k \in \mathbb{N}$. The k-negations fragment of the FO language of a signature σ is the set of all formulae having at most k negations $\neg$. Note that, following the grammar of FO languages provided above, this restriction also bounds the number of disjunctions and alternations between existential and universal quantifiers that formulae can have. Given a structure $\mathcal{A} = (A, \sigma, I)$, we are interested in the following problem:

k NEGATIONS SATISFIABILITY: given a formula Φ with at most k negations, decide $[\![\Phi]\!]_{\mathcal{A}} \neq \emptyset$.

3 Representations and parametrised complexity of signatures

Per se, a structure $\mathcal{A}$ cannot be analysed algorithmically, in particular because the elements of A do not have a notion of size. A standard way to resolve this issue is defining computability via the notion of representations (as it is done for instance in computable analysis [26]).

Representations. A *representation* for a set A is a surjective partial map $\rho :\subseteq \Sigma^* \to A$, where Σ is a finite alphabet. Words $w \in \Sigma^*$ are naturally equipped with a notion of *size*, i.e., their length, denoted by $|w|$. Not all words are valid representations for elements of A (ρ is partial) and each element from A can be represented in several ways (ρ is not assumed to be injective). Given a representation $\rho :\subseteq \Sigma^* \to A$, we write $(\approx_\rho) \subseteq \Sigma^* \times \Sigma^*$ for the equivalence relation $\{(w_1, w_2) : w_1, w_2 \in \mathrm{dom}(\rho) \text{ and } \rho(w_1) = \rho(w_2)\}$ and define $h_\rho \colon \mathrm{dom}(\rho)/_{\approx_\rho} \to A$ to be the bijection satisfying $h_\rho([w]_{\approx_\rho}) = \rho(w)$, for every $w \in \mathrm{dom}(\rho)$. Here, $\mathrm{dom}(\rho)/_{\approx_\rho}$ is the set of all equivalence classes $[w]_{\approx_\rho}$ of words $w \in \mathrm{dom}(\rho)$.

It is often more practical to represent elements of A by objects that are more sophisticated than words in Σ^*, such as tuples, automata, graphs, etc. Taking these representations does not change the notion of computability or complexity, because they can be easily encoded as words (over a bigger alphabet, if necessary). In our setting, of particular interest are representations as tuples of words. The notion of size for words trivially extends to tuples: $|(w_1, \ldots, w_n)| \coloneqq n + \sum_{i=1}^{n} |w_i|$. Given representations $\rho :\subseteq \Sigma^* \to A$ and $\rho' :\subseteq \Pi^* \to A'$, we rely on the following operations on representations:

- The Cartesian product $\rho \times \rho'$ of representations, defined as in Section 2.
- The representation $\mathrm{seq}(\rho) :\subseteq \mathrm{seq}(\Sigma^*) \to \mathrm{seq}(A)$ returning $(\rho(w_1), \ldots, \rho(w_n))$ on tuples $(w_1, \ldots, w_n) \in \mathrm{dom}(\rho)^n$, for every $n \in \mathbb{N}$.
- Given a map $\oplus \colon S \times S \to S$ with $S \supseteq A$, the representation $\mathrm{fold}[\oplus](\rho)$ recursively defined as $\mathrm{fold}[\oplus](\rho)(()) \coloneqq \emptyset$, and $\mathrm{fold}[\oplus](\rho)((w_1, \ldots, w_n)) \coloneqq \rho(w_1) \oplus \mathrm{fold}[\oplus](\rho)((w_2, \ldots, w_n))$.

We also require representations for basic objects such as $\mathbb{N}$, $\mathbb{Z}$ and so on. Specifically, we assume to have *canonical representations* ν_X for the following countable domains X:

- $X = \mathbb{N}$ or $X = \mathbb{Z}$, so that ν_X denote a representation of $\mathbb{N}$ or $\mathbb{Z}$, respectively.
- X is any finite set, e.g., we assume to have a representation $\nu_{\mathbb{B}}$ for the Booleans $\mathbb{B} = \{\top, \bot\}$,
- $X = \Sigma^*$ where Σ is any finite alphabet. In this case, $\nu_{\Sigma^*} \coloneqq \mathrm{id}_{\Sigma^*}$.

Implementations and computability. Let $\rho :\subseteq \Pi^* \to A$ and $\rho_1, \ldots, \rho_n$ be representations, with $\rho_i :\subseteq \Sigma_i^* \to A_i$. A function $f \colon A_1 \times \cdots \times A_n \to A$ is said to be $(\rho_1 \times \cdots \times \rho_n, \rho)$-*computable* if there is a computable function (in the usual sense of Turing machines) $F \colon \mathrm{dom}(\rho_1) \times \cdots \times \mathrm{dom}(\rho_n) \to \mathrm{dom}(\rho)$ such that $\rho(F(w_1, \ldots, w_n)) = f(\rho_1(w_1), \ldots, \rho_n(w_n))$ for all $w_i \in \mathrm{dom}(\rho_i)$, $i \in [1, n]$. The function F is said to be a $(\rho_1 \times \ldots \times \rho_n, \rho)$-*implementation* of f. It is convenient to avoid mentioning the representations of a computable function when it operates on canonical types. Given sets $A, A_1, \ldots, A_n$ admitting canonical representations, a function $f \colon A_1 \times \cdots \times A_n \to A$ is said to be *computable* whenever it is $(\nu_{A_1} \times \ldots \times \nu_{A_n}, \nu_A)$-*computable* ($\nu_{A_i}$ and ν_A are the canonical representations of A_i and A).

Let $\mathcal{A} = (A, \sigma, I)$ be a structure and $\rho :\subseteq \Sigma^* \to A$ be a representation. Let $\mathcal{M} \coloneqq (\mathrm{dom}(\rho), \sigma, J)$ be a structure where the interpretation function J associates computable functions to each function, family of functions and relations in σ, and makes $\approx_\rho$ a congruence for $\mathcal{M}$. The structure $\mathcal{M}$ is said to be a *ρ-implementation* of $\mathcal{A}$ whenever ρ is a homomorphism between $\mathcal{M}$ and $\mathcal{A}$. We highlight the fact that, compared to a standard homomorphism between structures, an implementation is always surjective (since ρ is surjective) and forces J to give an interpretation to functions and relations in σ in terms of computable functions.

▶ **Example 2.** The addition function $+\colon \mathbb{Z}^2 \to \mathbb{Z}$ is $(\nu_{\mathbb{Z}}^2, \nu_{\mathbb{Z}})$-computable. Since $\nu_{\mathbb{Z}}$ is the canonical representation of $\mathbb{Z}$, we simply say that $+$ is *computable*. This is the standard notion of computability over $\mathbb{Z}$. The structure $(\mathrm{dom}(\nu_{\mathbb{Z}}), +)$ is a $\nu_{\mathbb{Z}}$-implementation of $(\mathbb{Z}, +)$.

Parametrised complexity of signatures. The framework we define in the next section requires the introduction of a notion of parametrised complexity for the signature of a structure (which we call a *UXP signature*) which we now formulate. First, let us recall the standard notion of UXP reduction from parametrised complexity theory [11, Chapter 15]. Let Γ and Π be two finite alphabets, and $D \subseteq \Gamma^*$. A *parameter function* is a map $\eta\colon \Gamma^* \to \mathbb{N}$ such that $\eta(w) \geq 1$ for every $w \in \Gamma^*$. A computable function $F\colon D \to \Pi^*$ is said to be a *uniform slicewise polynomial reduction* for two parameter functions η and θ, or (η, θ)-UXP reduction in short, whenever there is an increasing map $G\colon \mathbb{N} \to \mathbb{N}$ such that for every $w \in \Gamma^*$, $F(w)$ runs in time $|w|^{G(\eta(w))}$ (w.l.o.g. assume $|w| \geq 2$) and $\theta(F(w)) \leq G(\eta(w))$.

As usual in computability theory, functions F with multiple arguments are handled by introducing a special symbol to the alphabet Γ, say $\#$, to separate the arguments, thus seeing F as a function in one input. For instance, an operator $\oplus\colon \Sigma_1^* \times \Sigma_2^* \to \Sigma^*$ can be interpreted by a computable function taking as inputs words $w_1 \# w_2$ with $(w_1, w_2) \in \Sigma_1^* \times \Sigma_2^*$. The product $(\eta_1 \cdot \eta_2)(w_1 \# w_2) \coloneqq \eta_1(w_1) \cdot \eta_2(w_2)$ of parameter functions $\eta_1\colon \Sigma_1^* \to \mathbb{N}$ and $\eta_2\colon \Sigma_2^* \to \mathbb{N}$ can be used to refine the complexity analysis of $\oplus$ to each of its two arguments. We write $\mathbf{1}$ for the trivial parameter function defined as $\mathbf{1}(w) \coloneqq 1$ for all $w \in \Sigma^*$.

Let $\mathcal{A} = (A, \sigma, I)$ be a structure, $\sigma = (\mathcal{F}, \mathcal{G}, \mathcal{R}, ar)$, $\rho :\subseteq \Sigma^* \to A$ be a representation, and $\eta\colon \Sigma^* \to \mathbb{N}$ be a parameter function. We say that $\mathcal{A}$ has a (ρ, η)-*UXP signature* whenever there is an interpretation function J such that *(i)* $(\mathrm{dom}(\rho), \sigma, J)$ is a ρ-implementation of $\mathcal{A}$ and *(ii)* J associates a $(\eta^{ar(f)}, \eta)$-UXP reduction to every $f \in \mathcal{F}$, a $(\mathbf{1} \cdot \eta^{ar(g)}, \eta)$-UXP reduction to every $(g, X) \in \mathcal{G}$, and a $(\eta^{ar(R)}, \mathbf{1})$-UXP reduction to every $R \in \mathcal{R}$.

▶ **Example 3.** Consider the structure $(L, \cup, \cap, (\cdot)^c)$ where L is the set of all regular languages over a finite alphabet Σ and $\cup$, $\cap$, and $(\cdot)^c$ are the canonical operations of union, intersection and complementation of languages, respectively. This structure has a tractable signature for the representation of regular languages as deterministic finite automata (DFAs), as all operations can be implemented in PTIME on DFAs. However, it does not have a tractable signature for the representation of regular languages as non-deterministic finite automata (NFAs), because computing $(\cdot)^c$ on NFAs requires first to determinise the automaton.

As in the case of representations, it is often more practical to have parameter functions from objects that are more sophisticated than words. Given a parameter function $\theta\colon \Sigma^* \to \mathbb{N}$, in this paper we consider the following operations $\mathrm{len}(\theta)\colon \mathrm{seq}(\Sigma^*) \to \mathbb{N}$, $\max(\theta)\colon \mathrm{seq}(\Sigma^*) \to \mathbb{N}$ and $\mathrm{depth}(\theta)\colon \mathrm{seq}(\mathrm{seq}(\Sigma^*)) \to \mathbb{N}$ on parameter functions (below, $\mathbf{w} = (w_1, \ldots, w_n)$):

$$\mathrm{len}(\theta)(\mathbf{w}) \coloneqq \textstyle\sum_{i=1}^{n} \theta(w_i), \qquad \max(\theta)(\mathbf{w}) \coloneqq \max_{i=1}^{n} \theta(w_i), \qquad \mathrm{depth}(\theta) \coloneqq \mathrm{len}(\mathrm{len}(\theta)).$$

4 A framework for the fixed negation fragment of first-order theories

Fix a structure $\mathcal{A} = (A, \sigma, I)$ and consider $\mathrm{FO}(\mathcal{A}) \coloneqq (\llbracket \mathcal{A} \rrbracket_{\mathrm{FO}}, \bot, \top, \vee, \wedge, -, (\pi, \mathbf{I}), (\pi^{\forall}, \mathbf{I}), \leq)$. In this section, we describe a framework that can be employed to show that the k negation satisfiability problem for $\mathrm{FO}(\mathcal{A})$ is in PTIME. Part of our framework is generic, i.e., applies to any first-order theory, while other parts are necessarily specific to the theory under study. This section covers the former part and highlights the latters.

To understand the framework it is helpful to take a moment to consider how we can exploit the fact that we only consider formulae with a fixed number of negations. For simplicity, let us focus for the time being on *quantified Boolean formulae* (QBF) in prenex

form. A first key question is whether bringing the quantifier-free part Φ of a QBF formula in a particular normal form can be computationally beneficial. Of course, due to our restrictions, Φ can be brought into DNF in PTIME. However, because of quantifier alternation together with the unbounded number of conjunctions, choosing this normal form comes with several intricacies. Another option we might try is to put Φ into a form where all but a fixed amount of constraints are in Horn form, and then try to rely on the algorithm to solve quantified Horn Boolean satisfiability in PTIME [17]. This works for the Boolean case, but not for an arbitrary theory. For instance, the quantified Horn satisfiability problem for the FO theory of $\mathcal{Z} = (\mathbb{Z}, 0, 1, +, =)$, i.e. weak Presburger arithmetic, is already NP-hard for the alternation prefix $\exists\forall$ and 2 variables, as we briefly sketch in Appendix A, and NEXPTIME-hard in general [9]. It turns out that a suitable normal form for Φ is given by formulae of the form $\Phi_1 - (\Phi_2 - (\cdots - (\Phi_{k-1} - \Phi_k)))$, where each Φ_i is a negation-free formula in DNF, and $\Psi_1 - \Psi_2$ is the relative complementation $\Psi_1 \wedge \neg\Psi_2$. As we will see in this section, this atypical normal form (introduced by Hausdorff in [13] and called *difference normal form* in [14]) not only fully makes use of our restriction on the number of negations, but also exhibits nice properties in relation to quantification.

A second key question is what representation of $[\![\mathcal{A}]\!]_{\mathrm{FO}}$ works best for our purposes, as formulae might not be the right "data structure". Though the difference normal form already sets how to treat disjunctions and negations, we have the flexibility to vary the representation of conjunctions of atomic formulae. Let us be a bit more precise. Consider a domain $\mathrm{D} \subseteq [\![\mathcal{A}]\!]_{\mathrm{FO}}$ containing *at least* the sets $[\![\Psi]\!]_{\mathcal{A}}$, for all $\Psi \in \mathrm{AC}(\sigma)$. We define $\mathrm{un}(\mathrm{D})$ to be the closure of D under the disjunction $\vee$, and $\mathrm{diffnf}(\mathrm{D})$ to be the smallest set containing $\mathrm{un}(\mathrm{D})$ and being closed under relative complements $X - Y$, with $X \in \mathrm{un}(\mathrm{D})$ and $Y \in \mathrm{diffnf}(\mathrm{D})$. From the notion of difference normal form we conclude that $\{[\![\Phi]\!]_{\mathcal{A}} : \Phi \text{ quantifier free}\} \subseteq \mathrm{diffnf}(\mathrm{D})$. Now, given a representation $\rho :\subseteq \Sigma^* \to \mathrm{D}$, the partial functions $\mathrm{un}(\rho) :\subseteq \mathrm{seq}(\Sigma^*) \to \mathrm{un}(\mathrm{D})$ and $\mathrm{diffnf}(\rho) :\subseteq \mathrm{seq}(\mathrm{seq}(\Sigma^*)) \to \mathrm{diffnf}(\mathrm{D})$, defined as $\mathrm{un}(\rho) \coloneqq \mathrm{fold}[\vee](\mathrm{D})$ and $\mathrm{diffnf}(\rho) \coloneqq \mathrm{fold}[-](\mathrm{un}(\rho))$ are representations of $\mathrm{un}(\mathrm{D})$ and $\mathrm{diffnf}(\mathrm{D})$, respectively. When ρ is a representation encoding D as conjunctions of atomic formulae, then $\mathrm{diffnf}(\rho)$ is encoding $\mathrm{diffnf}(\mathrm{D})$ in the difference normal from [13]. The key point is that the representation ρ can be selected so that D is encoded as something other than formulae. For instance, for linear arithmetic theories, alternative encodings are given by automata [6] or geometrical objects [10]. In Section 6 we will use the latter. Of course, selecting representations other than formulae requires an efficient way of changing representation, as stressed in the forthcoming Proposition 4 (see the map F). One last observation: above, we introduced D so that it could include sets beyond $[\![\Psi]\!]_{\mathcal{A}}$, where $\Psi \in \mathrm{AC}(\sigma)$. Further sets might be required in order to make $\mathrm{diffnf}(\mathrm{D})$ closed under (universal) projection. For instance, in weak integer arithmetic, the formula $\exists y : x = 2 \cdot y$, stating that x is even, cannot be expressed with a quantifier-free formula, hence $[\![\exists y : x = 2 \cdot y]\!]_{\mathcal{Z}}$ must be added to D.

The following proposition formalises the observations done in the last two paragraphs. We recall that an algorithm is said to be in χ-UXP, for a parameter $\chi\colon \Sigma^* \to \mathbb{N}$, if it runs in time $|w|^{G(\chi(w))}$ for every $w \in \Sigma^*$, for some function $G\colon \mathbb{N} \to \mathbb{N}$ not depending on w. A decision problem is in χ-UXP whenever there is a χ-UXP algorithm solving that problem.

▶ **Proposition 4.** *Fix $k \in \mathbb{N}$. Assume the following objects to be defined:*

- *a representation ρ of $\mathrm{D} \coloneqq \bigcup_{n\in\mathbb{N}} \mathrm{D}_n$, where, for all $n \in \mathbb{N}$, $\mathrm{D}_n \subseteq \mathcal{P}(A^n)$ is such that $[\![\Psi]\!]_{\mathcal{A}} \in \mathrm{D}_n$ for every $\Psi \in \mathrm{AC}(\sigma)$ having maximum variable x_n,*
- *a (ξ, θ)-UXP reduction $F\colon \mathrm{AC}(\sigma) \to \mathrm{dom}(\rho)$ s.t. $(\rho \circ F)(\Psi) = [\![\Psi]\!]_{\mathcal{A}}$ for all $\Psi \in \mathrm{AC}(\sigma)$.*

If $\mathcal{D} = (\mathrm{diffnf}(\mathrm{D}), \bot, \top, \vee, \wedge, -, (\pi, \mathbf{I}), (\pi^\forall, \mathbf{I}), \leq)$ has $(\mathrm{diffnf}(\rho), \mathrm{depth}(\theta))$-UXP signature,

- *the k negations satisfiability problem for $\mathrm{FO}(\mathcal{A})$ is in ξ-UXP (PTIME, if $\xi = \mathbf{1}$), and*
- *there is a ξ-UXP (PTIME, if $\xi = \mathbf{1}$) algorithm that, given a formula Φ of $\mathrm{FO}(\mathcal{A})$ having at most k negations, returns X in $\mathrm{dom}(\mathrm{diffnf}(\rho))$ such that $\mathrm{diffnf}(\rho)(X) = [\![\Phi]\!]_{\mathcal{A}}$.*

By virtue of what we said above, Proposition 4 should not be surprising: the reduction F enables an efficient conversion from $\mathrm{AC}(\sigma)$ to elements in $\mathrm{dom}(\rho)$, and (since $\mathcal{D}$ is a structure) $\mathrm{diffnf}(\mathrm{D})$ is closed under all the operations in the signature and thus it is equal to $[\![\mathcal{A}]\!]_{\mathrm{FO}}$. Consequently, $\mathrm{FO}(\mathcal{A})$ has a $(\mathrm{diffnf}(\rho), \mathrm{depth}(\theta))$-UXP signature, and one can efficiently use $\mathrm{diffnf}(\rho)$ as a data structure to carry out the algorithm to decide satisfiability, by simply invoking the various UXP reductions implementing the functions and relations in $\mathcal{D}$. We remark that the sole purpose of the parameter θ is to factor in the parameter ξ, and can be thought as $\theta \coloneqq \mathbf{1}$ in the case of $\xi \coloneqq \mathbf{1}$ (i.e., the case yielding PTIME algorithms).

Whereas the choice of D and F highly depends on the FO theory at hand, we show that a significant portion of the work required to prove that $\mathcal{D}$ has the desired UXP signature can be treated in a general way, thanks to the notion of difference normal form. This "automation" can be seen as the core of our framework, which provides a minimal set of subproblems that are sufficient to conclude that $\mathcal{D}$ has a $(\mathrm{diffnf}(\rho), \mathrm{depth}(\theta))$-UXP signature. Below, we divide those subproblems into two steps, one for Boolean connectives and one for quantification. One significant result in this context is that negation can be treated in a general way.

▶ **Step 1** (Boolean connectives).

1. *Show that the structure* $(\mathrm{D}, \wedge, \leq)$ *has a* (ρ, θ)*-UXP signature.*
2. *Show that the structure* $(\mathrm{un}(\mathrm{D}), \leq)$ *has a* $(\mathrm{un}(\rho), \mathrm{len}(\theta))$*-UXP signature.*

Step 1 asks to provide algorithms for solving typical computational problems that are highly domain-specific: Item 1 considers the intersection and inclusion problems for elements of D, with respect to the representation ρ, whereas Item 2 deals with the inclusion problem for unions of elements in D, with respect to the representation $\mathrm{un}(\rho)$. In the case of unions, we highlight the parameter $\mathrm{len}(\theta)$ which fixes the length of the union.

Once Step 1 is established, we are able to show that the full Boolean algebra (including relative complementation) of $\mathrm{diffnf}(\mathrm{D})$ has a UXP signature that is suitable for Proposition 4.

▶ **Lemma 5.** *Under the assumption that Step 1 is established,* $(\mathrm{diffnf}(\mathrm{D}), \bot, \top, \vee, \wedge, -, \leq)$ *has a* $(\mathrm{diffnf}(\rho), \mathrm{depth}(\theta))$*-UXP signature.*

The proof of this lemma boils down to the definition of suitable UXP reductions implementing the binary operations $\wedge$, $\vee$ and $-$. We will give further insights on how this is achieved in Section 5, where we study algorithmic aspects of the difference normal form.

Moving forward, we now consider projection and universal projection. Again, the goal is to minimise the efforts needed to add support for these operations. In this sense, the decision to adopt the difference normal becomes now crucial. First, we need to introduce a variant of the universal projection which we call relative universal projection. Given $Z \subseteq A^m$, $X \subseteq A^n$ and $\mathbf{i} = (i_1, \ldots, i_k) \in \mathbf{I}$, the *relative universal projection* $\pi^{\forall}_Z(\mathbf{i}, X)$ of X with respect to Z is defined as follows (where $M \coloneqq \max(m, n)$):

$$\pi^{\forall}_Z(\mathbf{i}, X) \coloneqq \{(a_1, \ldots, a_M) \in A^M : \ \mathbf{a} \coloneqq (a_1, \ldots, a_m) \in \pi(\mathbf{i}, Z) \text{ and for all } \mathbf{b} \in A^k \text{ if } \mathbf{a}[\mathbf{i} \leftarrow \mathbf{b}] \in Z \text{ then } (a_1, \ldots, a_n)[\mathbf{i} \leftarrow \mathbf{b}] \in X\}.$$

Informally speaking, $\pi^{\forall}_Z(\mathbf{i}, X)$ acts as a universal projection for the part of X that lies inside Z. Note that one can retrieve the universal projection as $\pi^{\forall}(\mathbf{i}, X) = \pi^{\forall}_{\top}(\mathbf{i}, X)$.

The lemma below outlines a key "mutual distribution" property of projection and relative universal projection over relative complement. In the context of the difference normal form, this property allows us to disregard complementation when adding support for quantification.

▶ **Lemma 6.** *We have* $\pi(\mathbf{i}, X - Y) = \pi(\mathbf{i}, X) - \pi^{\forall}_X(\mathbf{i}, Y)$ *and* $\pi^{\forall}_Z(\mathbf{i}, X - Y) = \pi^{\forall}_Z(\mathbf{i}, X) - \pi(\mathbf{i}, Y)$*, for every* $X \subseteq A^n$*,* $Y \subseteq A^m$*,* $Z \subseteq A^r$*, and* $\mathbf{i} \in \mathbf{I}$*.*

For instance, $\pi(\mathbf{i}, W - (X - (Y - Z))) = \pi(\mathbf{i}, W) - (\pi^{\forall}_W(\mathbf{i}, X) - (\pi(\mathbf{i}, Y) - \pi^{\forall}_Y(\mathbf{i}, Z)))$.

Below, let us write $\dot{\pi}$ for the restriction of the projection operator π on inputs $(\mathbf{i}, X)$ where $X \in \mathrm{D}$, and write $\dot{\pi}^{\forall}_{\_}$ for the restriction of the relativised universal projection $\pi^{\forall}_{\_}$ on inputs $(Z, \mathbf{i}, X)$ where $Z \in \mathrm{D}$ and $X \in \mathrm{un}(\mathrm{D})$. Thanks to Lemma 6, adding to Step 1 the following step is sufficient to conclude that $\mathcal{D}$ has the UXP signature required by Proposition 4.

▶ **Step 2** (Projection and universal projection).
1. *Show that* $\dot{\pi}(\mathbf{i}, X) \in \mathrm{diffnf}(\mathrm{D})$*, for every* $X \in \mathrm{D}$ *and* $\mathbf{i} \in \mathbf{I}$.
2. *Show that* $\dot{\pi}^{\forall}_{Z}(\mathbf{i}, X) \in \mathrm{diffnf}(\mathrm{D})$*, for every* $Z \in \mathrm{D}$*,* $\mathbf{i} \in \mathbf{I}$ *and* $X \in \mathrm{un}(\mathrm{D})$.
3. *Show a* $(\mathbf{1} \cdot \theta, \mathrm{depth}(\theta))$*-UXP reduction that* $(\nu_{\mathbf{I}} \times \rho, \mathrm{diffnf}(\rho))$*-implements* $\dot{\pi}$.
4. *Show a* $(\theta \cdot \mathbf{1} \cdot \mathrm{len}(\theta), \mathrm{depth}(\theta))$*-UXP reduction that* $(\rho \times \nu_{\mathbf{I}} \times \mathrm{un}(\rho), \mathrm{diffnf}(\rho))$*-implements* $\dot{\pi}^{\forall}_{\_}$.

▶ **Lemma 7.** *Under the assumption that Steps 1 and 2 are established, the structure* $\mathcal{D}$ *from Proposition 4 has a* $(\mathrm{diffnf}(\rho), \mathrm{depth}(\theta))$*-UXP signature.*

5 Closing prelattices under relative complement

Let us go back to the notion of difference normal form, which again are formulae of the form $\Phi_1 - (\Phi_2 - (\cdots - \Phi_k))$, where each Φ_i is negation-free and in DNF. Our framework is based on the idea of using these syntactic chains of relative complementations, let us call them *decreasing sequences*, as a way of closing a structure (not just formulae) under complement. Doing this allows the domain D to not be necessarily closed under complement. Then, one natural question to ask is under which assumptions do structures admit a computable notion of decreasing sequences. We give an answer to this question by showing that any prelattice $\mathcal{A}$ that is well-founded *or* distributive has a well-defined algebra over decreasing sequences (SDS algebra). We study computational aspects of SDS algebras that allows us to establish Lemma 5. When $\mathcal{A}$ is both well-founded and distributive, we also show that the SDS algebra act as a completion of $\mathcal{A}$ under relative complement (this result is not required for Lemma 5).

Prelattices. We assume familiarity with the order theoretic definition of a *lattice*. A structure $\mathcal{A} = (A, \vee, \wedge, \leq)$ is said to be a *prelattice* whenever $\leq$ is a *preorder*, and the quotient structure $\mathcal{A}/_{\approx} := (A/_{\approx}, \vee/_{\approx}, \wedge/_{\approx}, \leq/_{\approx})$ of $\mathcal{A}$ under the congruence $(\approx) := (\leq \cap \leq^{-1})$ is a lattice. Roughly speaking, a prelattice is a lattice that may not satisfy antisymmetry, i.e., distinct elements of A are allowed to be equal under $\leq$. If $\leq/_{\approx}$ has a least element L, we often add to the signature of $\mathcal{A}$ a constant symbol $\bot$ interpreted as an element $a \in A$ such that $[a]_{\approx} = L$. As usual, $\mathcal{A}$ is *well-founded* if there are no infinite sequences of elements that are strictly decreasing with respect to $\leq$, and it is *distributive* whenever $a \wedge (b \vee c) \approx (a \wedge b) \vee (a \wedge c)$, for every $a, b, c \in A$. A (pre)lattice $(L, \leq)$ is said to be *closed under relative complement* whenever, for every $x, y \in L$ there is an element $[x \setminus y] \in L$ satisfying $y \wedge [x \setminus y] = \bot$ and $x \leq y \vee [x \setminus y]$. (Pre)lattices may not be closed under relative complement, see for example the three elements lattice $(\{\bot, p, \top\}, \leq)$ with $\bot < p < \top$.

Decreasing sequences and SDS algebras. Fix a prelattice $\mathcal{A} = (A, \bot, \vee, \wedge, \leq)$ that is well-founded or distributive. Let $\approx$ and $<$ be the equivalence relation and strict partial order induced by $\leq$, respectively. We write $\mathrm{SDS}(\mathcal{A})$ for the set of all *strictly decreasing sequences (SDS)* over A, i.e., those (possibly empty) finite tuples $(a_1, \ldots, a_n) \in \mathrm{seq}(A)$ satisfying $a_{i+1} < a_i$ for every $i \in [1, n-1]$. Let $X := (a_1, \ldots, a_n), Y := (b_1, \ldots, b_m) \in \mathrm{SDS}(\mathcal{A})$. Given $a \in A$ and $X' \in \mathrm{SDS}(\mathcal{A})$, we write $X = \langle a; X' \rangle$ whenever $a = a_1$ and $X' = (a_2, \ldots, a_n)$. We write $X \dot{\approx} Y$ whenever $n = m$ and $a_i \approx b_i$ for all $i \in [1, n]$, or $X, Y \in \{(), a : a \in A \text{ and } a \approx \bot\}$. We recursively define the *cons* operator $(:) \colon A \times \mathrm{SDS}(\mathcal{A}) \to \mathrm{SDS}(\mathcal{A})$ as follows:

$$a : X \coloneqq \begin{cases} a & \text{if } X \in \{(), b : b \in A \text{ and } b \approx \bot\} \\ (a, a_1, \ldots, a_n) & \text{else if } a_1 < a \\ a : \big((a \wedge a_1) : (a_2, \ldots, a_n)\big) & \text{else if } a \wedge a_1 < a \\ \bot & \text{else if } n = 1 \text{ (here, } a \leq a_1) \\ (a \wedge a_2) : (a_3, \ldots, a_n) & \text{otherwise} \end{cases}$$

Intuitively, on input (a, X), $(:)$ returns an SDS that represents the relative complement of a and X. With a simple induction on the length of X one can show that $(:)$ is well-defined.

The *SDS algebra* of $\mathcal{A}$ is the structure $(\mathrm{SDS}(\mathcal{A}), \varnothing, \curlyvee, \curlywedge, -, \preceq)$. In this structure, $\varnothing$ is the constant function returning $\bot^{\mathcal{A}}$ from $\mathcal{A} \subseteq \mathrm{SDS}(\mathcal{A})$, the (*inclusion*) $\preceq$ is defined as $X \preceq Y \overset{\text{def}}{\Leftrightarrow} (X - Y) \dot{\approx} \varnothing$, and the functions $\curlyvee$ (*union*), $\curlywedge$ (*intersection*) and $-$ (*difference*), having arity two, are interpreted following the mutually recursive definitions (where, whenever their length is non-zero, we assume $X = \langle a; X' \rangle$ and $Y = \langle b; Y' \rangle$):

$$X \curlywedge Y := \begin{cases} \varnothing & \text{if } X \dot{\approx} \varnothing \text{ or } Y \dot{\approx} \varnothing, \\ (a \wedge b) : (X' \curlyvee Y') & \text{else} \end{cases} \qquad X \curlyvee Y := \begin{cases} Y & \text{if } X \dot{\approx} \varnothing, \\ X & \text{else if } Y \dot{\approx} \varnothing, \\ a : (X' - Y) & \text{else if } b \leq a, \\ (a \vee b) : \big((X' \curlyvee Y') - ((a \wedge b) : (X' \curlywedge Y'))\big) & \text{else.} \end{cases}$$

$$X - Y := \begin{cases} X & \text{if } X \dot{\approx} \varnothing \text{ or } Y \dot{\approx} \varnothing, \\ a : (X' \curlyvee Y) & \text{else} \end{cases}$$

The definitions of $\curlywedge$, $\curlyvee$ and $-$ observe validities of elementary set theories, and follow the idea that $\langle a; X' \rangle$ should represent the element $a - X'$. For instance, the last line in the definition of $X - Y$, where $X = \langle a; X' \rangle$, relies on the set validity $(E - F) - G = E - (F \cup G)$.

▶ **Lemma 8.** *Suppose that $\mathcal{A}$ is well-founded or distributive. Then, the functions $\curlyvee, \curlywedge, - :$ $\mathrm{SDS}(\mathcal{A}) \times \mathrm{SDS}(\mathcal{A}) \to \mathrm{SDS}(\mathcal{A})$ and $\preceq : \mathrm{SDS}(\mathcal{A}) \times \mathrm{SDS}(\mathcal{A}) \to \mathbb{B}$ are well-defined.*

For $\mathcal{A}$ well-founded, this lemma is proven by induction on the lexicographic ordering built from $\leq^{\mathcal{A}}$ and the total ordering on $\mathbb{N}$. For $\mathcal{A}$ distributive, one notes that the closure C of a finite set of elements $\{\bot, a_1, \ldots, a_n\} \subseteq A$ under $\vee$ and $\wedge$ is a prelattice that is finite up to $\approx$, making $\leq^{\mathcal{A}}/_{\approx}$ a well-founded relation when restricted to elements in C. These elements are the only ones computed by $\curlyvee$, $\curlywedge$ and $-$. The lemma then follows as in the well-founded case.

The following proposition clarifies the intention behind SDS algebras.

▶ **Proposition 9.** *Any well-founded and distributive lattice $\mathcal{A}$ embeds in $(\mathrm{SDS}(\mathcal{A}), \varnothing, \curlyvee, \curlywedge, \preceq)$, which is a well-founded distributive prelattice closed under relative complement.*

Showing that $\mathcal{A}$ embeds in $(\mathrm{SDS}(\mathcal{A}), \varnothing, \curlyvee, \curlywedge, \preceq)$ is simple. To show that the latter structure is closed under relative complement we rely on Birkhoff's representation theorem, a theorem that allows to construct set algebras isomorphic to $\mathcal{A}$. Birkhoff's representation theorem is usually given for *finite* distributive lattices. However, an inspection of its proof shows that finiteness can be replaced with well-foundedness in a simple way.

▶ **Theorem 10** ([3])**.** *Let $\mathcal{A} = (A, \bot, \vee, \wedge, \leq)$ be a well-founded distributive lattice. There is an algebra of sets $\mathcal{B} = (B, \emptyset, \cup, \cap, \subseteq)$ that is isomorphic to $\mathcal{A}$.*

Let $\mathcal{N} = (N, \emptyset, \cup, \cap, \setminus, \subseteq)$ be the algebra of sets obtained by closing the structure $\mathcal{B}$ above under the set difference $\setminus$. Thanks to Theorem 10, to show the closure under relative complement required by Proposition 9 it suffices providing a surjective homomorphism from the SDS algebra of $\mathcal{B}$ to the structure $\mathcal{N}$. The map $h : \mathrm{SDS}(\mathcal{B}) \to N$ defined as $h(X) \coloneqq \emptyset$ for $X \dot{\approx} \varnothing$, and otherwise as $h(X) \coloneqq a \setminus h(X')$ for $X = \langle a; X' \rangle$, is such a homomorphism.

UXP signatures for SDS algebras. We move to the computational aspects of SDS algebras. Let $\mathcal{A} = (A, \bot, \vee, \wedge, \leq)$ be a distributive (not necessarily well-founded) prelattice and $\rho :\subseteq \Sigma^* \to A$ be a representation. Let $\mathrm{SDS}(\rho) :\subseteq \mathrm{seq}(\Sigma^*) \to \mathrm{SDS}(\mathcal{A})$ be the representation of $\mathrm{SDS}(\mathcal{A})$ defined as $\mathrm{SDS}(\rho)(w_1, \dots, w_n) \coloneqq \rho(w_1) : \big(\cdots : (\rho(w_n))\big)$ for every $n \in \mathbb{N}$ and $w_1, \dots, w_n \in \mathrm{dom}(\rho)$, and undefined otherwise. Below, $\mathrm{len}(\eta)$ is defined as in Section 3.

▶ **Theorem 11.** *Let $\mathcal{A} = (A, \bot, \vee, \wedge, \leq)$ be a distributive prelattice with a (ρ, η)-UXP signature. The SDS algebra of $\mathcal{A}$ has a $(\mathrm{SDS}(\rho), \mathrm{len}(\eta))$-UXP signature.*

For the proof of this theorem, one considers the ρ-implementation $\mathcal{R}$ of $\mathcal{A}$ in which the functions $\vee$ and $\wedge$ are $(\eta \cdot \eta, \eta)$-UXP reductions and the relation $\leq$ is a $(\eta \cdot \eta, \mathbf{1})$-UXP reduction. The structure $\mathcal{R}$ is a distributive prelattice, hence it has a well-defined SDS algebra $(\mathrm{SDS}(\mathcal{R}), \varnothing, \curlyvee, \curlywedge, -, \preceq)$. By following the above definitions of $\curlyvee, \curlywedge, -$ and $\preceq$, one can provide $(\mathrm{len}(\eta)^2, \mathrm{len}(\eta))$-UXP and $(\mathrm{len}(\eta) \cdot \mathrm{len}(\eta), \mathbf{1})$-UXP reductions for the functions and relations of the structure $\mathcal{S} = (\mathrm{dom}(\mathrm{SDS}(\rho)), \varnothing, \curlyvee, \curlywedge, -, \preceq)$ that $\mathrm{SDS}(\rho)$-implements the SDS algebra of $\mathcal{A}$.

Theorem 11 gives us what we need to prove Lemma 5. Step 1 of the framework implies that $\mathcal{U} = (\mathrm{un}(\mathrm{D}), \bot, \vee, \wedge, \leq)$ is a distributive prelattice with a $(\mathrm{un}(\rho), \mathrm{len}(\theta))$-UXP signature. By Theorem 11, the SDS algebra of $\mathcal{U}$ has a $(\mathrm{SDS}(\mathrm{un}(\rho)), \mathrm{depth}(\theta))$-UXP signature. So, Lemma 5 follows as one provides a surjective homomorphism from the SDS algebra of $\mathcal{U}$ to $(\mathrm{diffnf}(\mathrm{D}), \bot, \top, \vee, \wedge, -, \leq)$. This surjective homorphism is obtained by updating map h used in the proof of Proposition 9 so that it uses the operator $-$ instead of the set difference $\setminus$.

6 Weak linear integer arithmetic

In this section, we briefly discuss how to instantiate the framework of Section 4 to weak Presburger arithmetic (weak PA), i.e. the FO theory of the structure $\mathcal{Z} = (\mathbb{Z}, 0, 1, +, =)$.

Setup. According to Proposition 4, instantiating the framework requires first to define the domain D, its representation ρ and the change of representation $F\colon \mathrm{AC}(\sigma) \to \mathrm{dom}(\rho)$. In weak PA, conjunctions of atomic formulae are systems of affine equations, which over $\mathbb{Z}$ define shifted integer lattices (SL), which are not necessarily fully dimensional. We let D_n be the set of all shifted (integer) lattices of $\mathbb{Z}^n$, so that D is the set of all shifted lattices of $\mathbb{Z}^n$ for some n. We represent elements in D with the standard representation of a SL as a base point and an independent periodic set. Recall that we write $\nu_{\mathbb{Z}^n}$ for the canonical representation of $\mathbb{Z}^n$ (see Section 3). Formally, for every $n \in \mathbb{N}$, if v_0 represents a vector in $\mathbb{Z}^n$, and $v_1, \dots, v_k$ represent linearly independent vectors in $\mathbb{Z}^n$, then we let

$$\rho_{\mathrm{SL}}(n, v_0, \cdots, v_k) \coloneqq \nu_{\mathbb{Z}^n}(v_0) + \mathrm{span}_{\mathbb{Z}}\{\nu_{\mathbb{Z}^n}(v_1), \cdots, \nu_{\mathbb{Z}^n}(v_k)\}.$$

A PTIME function F allowing to change representation from conjunctions of atomic formulae to elements in $\mathrm{dom}(\rho_{\mathrm{SL}})$ can be obtained thanks to the following well-known algorithm.

▶ **Proposition 12** ([16]). *There is a PTIME algorithm to compute the Hermite normal form, along with the transformation matrices, of a given matrix with integer entries.*

Since F runs in PTIME, the parameter ξ in Proposition 4 equals $\mathbf{1}$, and we need to show that $\mathcal{D}$ has a $(\mathrm{diffnf}(\rho_{\mathrm{SL}}), \mathrm{depth}(\theta))$-UXP signature for $\theta \coloneqq \mathbf{1}$, by establishing Steps 1 and 2.

Step 1. Both the problems of computing intersections and testing inclusions for two shifted lattices represented as in ρ_{SL} reduces to solving systems of linear equations over $\mathbb{Z}$, which can be done in PTIME again thanks to Proposition 12. This establishes Item 1 of Step 1.

Item 2 asks for an algorithm to solve inclusion between union of shifted lattices. This is a computationally expensive operation. Roughly speaking, one first notices that it suffices to be able to test $Z \leq X$ where $Z \subseteq \mathbb{Z}^k$ is a SL and $X := \bigvee_{i \in I} X_i \subseteq \mathbb{Z}^k$, where all X_i are SL. This inclusion testing can be performed by checking that $|Z \cap [0,d)^k| = |(Z \wedge X) \cap [0,d)^k|$ for a well-chosen value of $d \in \mathbb{N}$. Hence, inclusion reduces to a counting problem for lattice points in a box, which in turn reduces to computing lattice determinants:

▶ **Lemma 13.** *Let $L \subseteq \mathbb{Z}^k$ be a lattice and $d > 0$ such that $d\mathbb{Z}^k \subseteq L$, then $|L \cap [0,d)^k| = \frac{d^k}{\det(L)}$.*

To extend the above lemma to union of shifted lattices, we rely on an inclusion-exclusion formula which requires the algorithm to consider all possible $2^{|I|}$ intersections between the X_i. This leads to a procedure that is exponential in the number of elements in the union, which is however sufficient to establish Item 2, since it asks for an algorithm that runs in PTIME when the length of the union is fixed; see the parameter $\mathrm{len}(\theta)$.

▶ **Lemma 14.** *There is an algorithm that given $z \in \mathrm{dom}(\rho_{\mathrm{SL}})$ and $x \in \mathrm{dom}(\mathrm{un}(\rho_{\mathrm{SL}}))$ decides $\rho_{\mathrm{SL}}(z) \leq \mathrm{un}(\rho_{\mathrm{SL}})(x)$ in time $2^{\mathrm{len}(\mathbf{1})(x)}\mathrm{poly}(|x|,|z|)$. In particular, Item 2 of Step 1 holds.*

Step 2. Establishing Items 1 and 3 is simple: thanks to our choice of representation, i.e. ρ_{SL}, given $X \in \mathrm{dom}(\rho_{\mathrm{SL}})$ and $\mathbf{i} \in \mathbf{I}$, $\pi(\mathbf{i}, X)$ can be computed by simply crossing out the entries of all vectors of X corresponding to the indices in $\mathbf{i}$. On the contrary, the algorithm for universal projections $\pi^{\forall}_Z X$ required by Items 2 and 4 turns out to be challenging to compute. Intuitively, similarly to inclusion testing, we need to count points in unions of SL but in a parametric way. This means that given $X = \bigvee_{\ell \in L} X_\ell$, where all X_ℓ are SL, every intersection $\bigwedge_{j \in J} X_j$ ($J \subseteq L$) in the inclusion-exclusion formula may or may not need to be counted, depending on the value of a parameter f belonging of a certain set of parameters $\mathcal{F}$ (see lemma below). The algorithm therefore considers all possible ways in which intersections may or may not be taken, which is roughly $2^{2^{|L|}}$. This allows us to conclude a rather surprising fact: the relative universal projection can be expressed as a complex combination of unions, intersections, projections and relative complementations that are exclusively applied to the initial sets in input. The number of these operations only depend on the length $|L|$.

▶ **Lemma 15.** *Consider $X = \bigvee_{\ell \in L} X_\ell$, where $X_\ell \in \mathrm{D}$ for all $\ell \in L$, and let $Z \in \mathrm{D}$ and $\mathbf{i} \in \mathbf{I}$. Then, there is a set of parameters $\mathcal{F} \subseteq (2^L \to \{0,1\})$ such that*

$$\pi^{\forall}_Z(\mathbf{i}, X) = \bigvee_{f \in \mathcal{F}} \Big(\Big(\bigwedge_{J : f(J)=1} \bigwedge_{j \in J} \pi(\mathbf{i}, X_j \wedge Z) \Big) - \Big(\bigvee_{J : f(J)=0} \bigwedge_{j \in J} \pi(\mathbf{i}, X_j \wedge Z) \Big) \Big).$$

Given $z, (x_\ell)_{\ell \in L} \in \mathrm{dom}(\rho_{\mathrm{SL}})$ s.t. $\rho_{\mathrm{SL}}(z) = Z$ and $\rho_{\mathrm{SL}}(x_\ell) = X_\ell$, the set $\mathcal{F}$ can be computed in time $\mathrm{poly}(|z|, \max_{\ell \in L}(|x_\ell|), 2^{2^{|L|}+|L|})$. In particular, Items 2 and 4 of Step 2 hold.

To clarify, since the parameter $\mathrm{len}(\theta)$ in Item 4 fixes the length $|L|$, the right-hand side of the above equation only has a fixed number of operations, and can thus be evaluate efficiently thanks to the other steps of the framework. Then, by Lemma 7 and Proposition 4, we get:

▶ **Theorem 16.** *Fix $k \in \mathbb{N}$. The k negations satisfiability problems for weak PA is in PTIME.*

By Proposition 4 we also conclude that there is a PTIME procedure that given a formula Φ from $\mathrm{FO}(\mathcal{Z})$ with k negations returns an element of $\mathrm{dom}(\mathrm{diff}(\mathrm{un}(\rho_{\mathrm{SL}})))$ representing $[\![\Phi]\!]_{\mathcal{Z}}$.

7 Final remarks

We developed a framework to establish polynomial-time decidability of fixed negation sentences of first-order theories whose signatures enjoy certain (parametrised) complexity properties. A key feature of the framework is that it treats complementation in a general way, and considers universal projection as a first-class citizen. Note that, a priori, the latter operation might be easier than the former to decide, as shown for instance in [8].

We instantiated our framework to show that the fixed negation fragment of weak PA is in PTIME, in sharp contrast with (standard) PA [20]. Due to space constrains, we did not provide further instantiation of our framework. We know that it can be used to show that the fixed negation fragment of weak linear real arithmetic is in PTIME. We believe that the framework provides a sensible approach to study fixed negation fragments of FO extensions of, e.g., certain abstract domains. While the framework obviously works for interval arithmetic, the case of linear octagon arithmetic [19], whose full FO theory is PSPACE-complete [2], seems already non-trivial. More generally, since the various steps required to instantiate the framework only consider natural computational problems (inclusions and projections), our hope is to also tackle theories outside the world of arithmetic.

References

1 Eric Bach and Jeffrey Shallit. *Algorithmic Number Theory, Vol 1: Efficient Algorithms*. Foundations of Computing. MIT Press, 1996.

2 Michael Benedikt, Dmitry Chistikov, and Alessio Mansutti. The complexity of Presburger arithmetic with power or powers. In *ICALP*, 2023. To appear.

3 Garrett Birkhoff. Rings of sets. *Duke Math. J.*, 3(3):443–454, 1937. doi:10.1215/S0012-7094-37-00334-X.

4 Manuel Bodirsky, Barnaby Martin, Marcello Mamino, and Antoine Mottet. The complexity of disjunctive linear diophantine constraints. In *MFCS*, pages 33:1–33:16, 2018. doi:10.4230/LIPIcs.MFCS.2018.33.

5 Itshak Borosh and Leon B. Treybing. Bounds on positive integral solutions of linear Diophantine equations. *Proc. Am. Math. Soc.*, 55:299–304, 1976. doi:10.2307/2041711.

6 J. Richard Büchi. Weak second-order arithmetic and finite automata. *Math. Logic Quart.*, 6(1-6):66–92, 1960. doi:10.1002/malq.19600060105.

7 Hubie Chen. A rendezvous of logic, complexity, and algebra. *ACM Comput. Surv.*, 42(1):2:1–2:32, 2009. doi:10.1145/1592451.1592453.

8 Dmitry Chistikov and Christoph Haase. On the complexity of quantified integer programming. In *ICALP*, pages 94:1–94:13, 2017. doi:10.4230/LIPIcs.ICALP.2017.94.

9 Dmitry Chistikov, Christoph Haase, Zahra Hadizadeh, and Alessio Mansutti. Higher-order quantified Boolean satisfiability. In *MFCS*, pages 33:1–33:15, 2022. doi:10.4230/LIPIcs.MFCS.2022.33.

10 Dmitry Chistikov, Christoph Haase, and Alessio Mansutti. Geometric decision procedures and the VC dimension of linear arithmetic theories. In *LICS*, pages 59:1–59:13, 2022. doi:10.1145/3531130.3533372.

11 Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, 1999. doi:10.1007/978-1-4612-0515-9.

12 Erich Grädel. Simple sentences that are hard to decide. *Inf. Comput.*, 94(1):62–82, 1991. doi:10.1016/0890-5401(91)90033-X.

13 Felix Hausdorff. *Grundzüge der Mengenlehre*. Veit and Company, Leipzig, 1914. [Accessed 7th August 2023]. URL: https://archive.org/details/grundzgedermen00hausuoft.

14 Markus Junker. A note on equational theories. *J. Symb. Log.*, 65(4):1705–1712, 2000. doi:10.2307/2695070.

15 Ravindran Kannan. Test sets for integer programs, ∀∃ sentences. *Polyhedral Combinatorics, Proc. of a DIMACS workshop*, pages 38–48, 1990.

16 Ravindran Kannan and Achim Bachem. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM J. Comput.*, 8(4):499–507, 1979. doi:10.1137/0208040.

17 Marek Karpinski, Hans Kleine Büning, and Peter H. Schmitt. On the computational complexity of quantified horn clauses. In *CSL*, pages 129–137, 1987. doi:10.1007/3-540-50241-6_34.

18 Hendrik W. Lenstra Jr. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983. doi:10.1287/moor.8.4.538.

19 Antoine Miné. The octagon abstract domain. *High. Order Symb. Comput.*, 19(1):31–100, 2006. doi:10.1007/s10990-006-8609-1.

20 Danny Nguyen and Igor Pak. Short Presburger arithmetic is hard. *SIAM J. Comput.*, 51(2):17:1–30, 2022. doi:10.1137/17M1151146.

21 Mojżesz Presburger. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. In *Comptes Rendus du I congres de Mathematiciens des Pays Slaves*, pages 92–101, 1929.

22 Bruno Scarpellini. Complexity of subcases of Presburger arithmetic. *Trans. Am. Math. Soc.*, 284:203–218, 1984. doi:10.2307/1999283.

23 Uwe Schöning. Complexity of presburger arithmetic with fixed quantifier dimension. *Theory Comput. Syst.*, 30(4):423–428, 1997. doi:10.1007/s002240000059.

24 Larry J. Stockmeyer. The polynomial-time hierarchy. *Theor. Comput. Sci.*, 3(1):1–22, 1976. doi:10.1016/0304-3975(76)90061-X.

25 Joachim von zur Gathen and Malte Sieveking. A bound on solutions of linear integer equalities and inequalities. *Proc. Am. Math. Soc.*, 72(1):155–158, 1978. doi:10.1080/00029890.1978.11994639.

26 Klaus Weihrauch. *Computable Analysis*. Springer, 2000. doi:10.1007/978-3-642-56999-9.

A Hardness of quantified weak PA Horn formulas

In this appendix, we complement the tractability result for the k-negation fragment of weak PA established in Section 6 with an NP lower bound for (quantified) weak PA Horn formulas. This lower bound only requires two variables x and y, where x is quantified existentially and y is quantified universally. The matrix of such a 2-variable formula is of the form

$$\bigwedge_{1\leq i\leq n} \big(a_i \cdot x + b_i \cdot y = c_i \to a'_i \cdot x + b'_i \cdot y = c'_i\big).$$

We show this result by a reduction from the problem of deciding a univariate system of non-congruences $\bigwedge_{i=1}^{k} x \not\equiv r_i \pmod{m_i}$, where $m_i \geq 2$ and $r_i \in [0, m_i - 1]$ for every $i \in [1, k]$. This problem is shown NP-hard in [1, Theorem 5.5.7]. The reduction directly follows from the following equivalence: for every $x \in \mathbb{Z}$,

$$\bigwedge_{i=1}^{k} x \not\equiv r_i \pmod{m_i} \iff \forall y : \bigwedge_{i=1}^{k} \big(x - r_i = m_i \cdot y \to y = 3 \cdot x + 1\big).$$

First, consider $x \in \mathbb{Z}$ satisfying the left-hand side. Pick $y \in \mathbb{Z}$. We have $x - r_i \neq m_i \cdot y$ for every $i \in [1, k]$. So, every antecedent of the implications in $\bigwedge_{i=1}^{k} \big(x - r_i = m_i \cdot y \to y = 3 \cdot x + 1\big)$ is false, showing the right-hand side. For the other direction, consider an $x \in \mathbb{Z}$ satisfying the right-hand side. It suffices to show that, for every $i \in [1, k]$ and $y \in \mathbb{Z}$, if $x - r_i = m_i \cdot y$ then $y \neq 3 \cdot x + 1$. This implies that, for the right-hand side to hold, it must be the case that $x - r_i \neq m_i \cdot y$ for every $i \in [1, k]$ and $y \in \mathbb{Z}$; proving the left-hand side. *Ad absurdum*, suppose that $x - r_i = m_i \cdot y$ and $y = 3 \cdot x + 1$ hold. Then, $x = -\frac{r_i + m_i}{3 \cdot m_i - 1}$. However, $0 < \frac{r_i + m_i}{3 \cdot m_i - 1} \leq \frac{2 \cdot m_i - 1}{3 \cdot m_i - 1} < 1$, as $m_i \geq 2$ and $r_i \in [0, m_i - 1]$, contradicting that x is an integer.

▶ **Proposition 17.** *Deciding $\exists\forall$ weak PA Horn sentences in two variables is NP-hard.*