Polynomial-Time Satisfiability Tests for Boolean Fragments of Set Theory*

Domenico Cantone[0000−0002−1306−1166]1, Andrea De Domenico[2], Pietro Maugeri[1], and Eugenio G. Omodeo[0000−0003−3917−1942]3

[1] Dept. of Mathematics and Computer Science, University of Catania, Italy
[2] Scuola Superiore di Catania, University of Catania, Italy
[3] Dept. of Mathematics and Geosciences, University of Trieste, Italy
{domenico.cantone,pietro.maugeri}@unict.it,
andrea.dedomenico@studium.unict.it, eomodeo@units.it

Abstract. We recently undertook an investigation aimed at identifying small fragments of set theory (which in most cases are sublanguages of Multi-Level Syllogistic) endowed with polynomial-time satisfiability decision tests, potentially useful for automated proof verification. Leaving out of consideration the membership relator $\in$ for the time being, in this note we provide a complete taxonomy of the polynomial and the NP-complete fragments involving, besides variables intended to range over the von Neumann set-universe, the Boolean operators $\cup, \cap, \setminus$, the Boolean relators $\subseteq, \not\subseteq, =, \neq$, and the predicates '$\cdot = \varnothing$' and '$\mathsf{Disj}(\cdot,\cdot)$', meaning 'the argument set is empty' and 'the arguments are disjoint sets', along with their opposites '$\cdot \neq \varnothing$' and '$\neg\mathsf{Disj}(\cdot,\cdot)$'.

Keywords: Satisfiability problem, Computable set theory, Boolean set theory, Proof verification, NP-completeness.

Introduction

The decision problem for fragments of set theory, namely the problem of establishing algorithmically for any formula φ in a given fragment whether or not φ is valid in the von Neumann universe of sets, has been thoroughly investigated over the last four decades within the field of *Computable Set Theory*. Research has mainly focused on the equivalent *satisfiability problem*, namely the problem of establishing in an effective manner, for any formula in a given fragment, whether an assignment of sets to its free variables exists that makes the formula true.

The initial goal (back in 1979) envisaged an automated proof verifier based on set theory, within which it would become possible to carry out an extensive formalization of classical mathematics. The inferential kernel of such a proof assistant should have embodied decision procedures intended to capture the 'obvious' (deduction steps). Very soon, and long before the proof verifier came into

* We gratefully acknowledge support from "Università degli Studi di Catania, Piano della Ricerca 2016/2018 Linea di intervento 2" and from the INdAM-GNCS 2019 research project "Logic Programming for early detection of pancreatic cancer".

existence, the initial goal sparked a foundational quest aimed at drawing the precise frontier between the decidable and the undecidable in set theory (and also in other important mathematical theories). This inspired much of the subsequent work. Several extensions of the progenitor fragments MLS and MLSS of set theory were proved to have a solvable satisfiability problem, which led to a substantial body of results, partly comprised in the monographs [3,5,16,15,8].

We recall that MLS (an acronym for *Multi-Level Syllogistic*, see [10]) copes with propositional combinations of literals of the form

$$x = y \cup z, \quad x = y \cap z, \quad x = y \setminus z, \quad x \in y; \tag{1}$$

to these constructs, MLSS adds the singleton operator $\{\cdot\}$.[4] Unfortunately, as shown in [4], the satisfiability problem for either MLS or MLSS is NP-complete, even if restricted to conjunctions of flat literals of type (1) plus negative literals of type $x \notin y$. All extensions of MLS will hence have, in their turn, an NP-hard satisfiability problem (in fact, even hyperexponential in some cases; see [1,6,7]). Notwithstanding, the decision algorithm for an enriched variant of MLSS, implemented along the guidelines of [9], has come to play a key role among the inference mechanisms available in the proof-checker ÆtnaNova, aka Ref [16]. In view of the pervasiveness of that mechanism in actual uses of ÆtnaNova (as discussed, e.g., in [14, Sect. 3] and in [15, Sect. 5.3.1]), it will pay off to circumvent whenever possible the poor performances occasionally originating from the full-strength decision algorithm.

This is why we recently undertook an investigation aimed at identifying useful 'small' fragments of set theory (which in most cases are subfragments of MLS) endowed with polynomial-time decision tests.

In this note we report on results focused, for the time being, on fragments that exclude the membership relator $\in$.[5] We provide a complete taxonomy of the polynomial and the NP-complete fragments involving, besides set variables intended to range over the von Neumann universe of all sets (see below), the Boolean operators $\cup, \cap, \setminus$ and relators $\subseteq, \not\subseteq, =, \neq$, and the predicates (both affirmed and negated) '$\cdot = \varnothing$' and '$\mathsf{Disj}(\cdot, \cdot)$', expressing respectively that a specified set is empty and that two specified sets are disjoint.

The paper is organized as follows: Sect. 1 introduces the syntax and semantics of a language in which several hundreds of decidable fragments will be framed in this note; a subsection of it defines 'expressibility', a notion which

[4] Note that, thanks to the following equivalences, $\neq$ and $\notin$ are available in MLS and $\in$ is eliminable from MLSS:

$$\begin{aligned} &x \neq y \leftrightarrow \exists w\,(\, x \in w \ \wedge\ y \notin w\,)\,,\\ &y \notin w \leftrightarrow \exists v\,(\, y \in v \ \wedge\ w \cap v = v \setminus v\,)\,,\\ &x \in w \leftrightarrow \{x\} \cap w = \{x\}\,. \end{aligned}$$

Also note that by adding Cartesian square literals $x = y \times y$ and cardinality literals $|x| = |y|$, $|x| \neq |y|$ to MLS, one makes the satisfiability problem undecidable [2].

[5] Some preliminary results involving the membership relator $\in$ will also be reviewed in Appendix B.

eases the systematic assessment of the complexities of the satisfiability decision tests. A complexity-based classification of the fragments under consideration is highlighted in Sect. 2. The paper terminates with a conclusion and some hints for future research, and with two appendices containing, respectively, the proofs of three lemmas on expressibility matters, and a brief overview of the complexity taxonomy for a small fragment involving the membership relator.

1 Boolean set theory

This section introduces an interpreted language regarding sets, whose acronym $\mathbb{BST}$ stands for 'Boolean set theory'. The constructs of $\mathbb{BST}$ are borrowed from the algebraic theory of Boolean rings (see [12]), but its variables are meant to range over a universe of nested (as opposed to 'flat') sets. We dub $\mathbb{BST}$ a 'theory' simply to emphasize that it has a decidable satisfiability problem. Below we will browse a wide range of subproblems of the satisfiability problem referring to the whole of $\mathbb{BST}$, and will assess the algorithmic complexity of those subproblems.

We postpone to future reports the treatment of $\in$, the membership relation. Adding $\in$ to $\mathbb{BST}$ does not disrupt its decidability and truly calls for nested sets.

1.1 Syntax

The fragments of set theory investigated within the project we are reporting about are parts, syntactically delimited, of a specific quantifier-free language

$$\mathbb{BST} := \mathsf{BST}(\cup, \cap, \setminus, =\varnothing, \neq\varnothing, \mathsf{Disj}, \neg\mathsf{Disj}, \subseteq, \not\subseteq, =, \neq).$$

This is the collection of all conjunctions of literals of the types

$$\begin{array}{cccc} s = \varnothing, & s \neq \varnothing, & \mathsf{Disj}(s,t), & \neg\mathsf{Disj}(s,t), \\ s \subseteq t, & s \not\subseteq t, & s = t, & s \neq t, \end{array}$$

where s and t stand for terms assembled from a denumerably infinite supply of set variables $x, y, z, \ldots$ by means of the Boolean set operators: union $\cup$, intersection $\cap$, and set difference $\setminus$.

More generally, we shall denote by $\mathsf{BST}(\mathsf{op}_1, \ldots, \mathsf{pred}_1, \ldots)$ the subtheory of $\mathbb{BST}$ involving only the set operators $\mathsf{op}_1, \ldots$ (drawn from the set $\{\cup, \cap, \setminus\}$) and the predicate symbols $\mathsf{pred}_1, \ldots$ (drawn from $\{=\varnothing, \neq\varnothing, \mathsf{Disj}, \neg\mathsf{Disj}, \subseteq, \not\subseteq, =, \neq\}$).

1.2 Semantics

For any $\mathbb{BST}$-conjunction φ, we shall denote by $\mathit{Vars}(\varphi)$ the collection of set variables occurring in φ; $\mathit{Vars}(\tau)$ is defined likewise, for any $\mathbb{BST}$-term τ.

A *set assignment* M is any function sending a collection of set variables V (called the *domain* of M and denoted $\mathsf{dom}(M)$) into the von Neumann universe $\mathcal{V}$ of well-founded sets. We recall that the *von Neumann universe* (see [13, pp.95–102]), aka *von Neumann cumulative hierarchy*, is built up in stages as the union

$\mathcal{V} \coloneqq \bigcup_{\alpha \in On} \mathcal{V}_\alpha$ of the levels $\mathcal{V}_\alpha \coloneqq \bigcup_{\beta<\alpha} \mathscr{P}(\mathcal{V}_\beta)$, with α ranging over the class On of all ordinal numbers, where $\mathscr{P}(\cdot)$ is the powerset operator.

Natural designation rules attach recursively a value to every term τ of $\mathbb{BST}$ such that $\mathit{Vars}(\tau) \subseteq \mathsf{dom}(M)$, for any set assignment M; here is how:

$$M(s \cup t) \coloneqq Ms \cup Mt, \quad M(s \cap t) \coloneqq Ms \cap Mt, \quad \text{and} \quad M(s \setminus t) \coloneqq Ms \setminus Mt.$$

We also put

$$M(s = \varnothing) \coloneqq \begin{cases} \mathsf{true} & \text{if } Ms = \emptyset \\ \mathsf{false} & \text{otherwise,} \end{cases} \qquad M(\mathsf{Disj}(s,t)) \coloneqq \begin{cases} \mathsf{true} & \text{if } Ms \cap Mt = \emptyset \\ \mathsf{false} & \text{otherwise,} \end{cases}$$
$$M(s \neq \varnothing) \coloneqq \neg M(s = \varnothing) \qquad M(\neg\mathsf{Disj}(s,t)) \coloneqq \neg M(\mathsf{Disj}(s,t))$$

for all literals $s = \varnothing$, $s \neq \varnothing$, $\mathsf{Disj}(s,t)$, and $\neg\mathsf{Disj}(s,t)$ of $\mathbb{BST}$ (and similarly for the literals of $\mathbb{BST}$ of the remaining types $s \subseteq t$, $s \not\subseteq t$, $s = t$, and $s \neq t$), and then, recursively,

$$M(\varphi \wedge \psi) \coloneqq M\varphi \wedge M\psi$$

when φ, ψ are $\mathbb{BST}$-conjunctions, $\mathit{Vars}(\varphi) \subseteq \mathsf{dom}(M)$, and $\mathit{Vars}(\psi) \subseteq \mathsf{dom}(M)$.

Given a conjunction φ and a set assignment M such that $\mathit{Vars}(\varphi) \subseteq \mathsf{dom}(M)$, we say that M *satisfies* φ, and write $M \models \varphi$, if $M\varphi = \mathsf{true}$. When M satisfies φ, we also say that M is a *model* of φ.

A conjunction φ is said to be *satisfiable* if it has some model, else *unsatisfiable*.

$\mathbb{BST}$ is a sublanguage of MLS, which has an NP-complete satisfiability problem [4]; since, in their turn, the fragments of set theory which we shall examine are included in $\mathbb{BST}$, their satisfiability problems belong to NP.

1.3 Expressibility

The technique of reduction has been our main tool in the construction of the complexity taxonomy of $\mathbb{BST}$-fragments, which will be presented at length in Sect. 2.

In our case, reductions have been mostly based on the standard notion of 'context-free' expressibility:

Definition 1 (Expressibility). *A formula $\psi(\boldsymbol{x})$ is* expressible *in a fragment $\mathcal{T}$ of $\mathbb{BST}$, if there exists a $\mathcal{T}$-conjunction $\Psi(\boldsymbol{x}, \boldsymbol{y})$ such that*

$$\models \psi(\boldsymbol{x}) \longleftrightarrow (\exists \boldsymbol{y})\, \Psi(\boldsymbol{x}, \boldsymbol{y}),$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ stand for tuples of set variables.

We also devised a more general notion of 'context-sensitive' expressibility, also characterized by its complexity. We named it $\mathcal{O}(f)$-expressibility, where $f \colon \mathbb{N} \to \mathbb{N}$ is any mapping intended to bound the complexity of the underlying rewriting procedure.

Definition 2 ($\mathcal{O}(f)$-expressibility). *Let $\mathcal{T}$ be a fragment of $\mathbb{BST}$ and $f\colon \mathbb{N} \to \mathbb{N}$ a given mapping. A formula $\psi(\boldsymbol{x})$ is $\mathcal{O}(f)$-*expressible *in $\mathcal{T}$ if there exists a mapping*

$$\varphi(\boldsymbol{y}) \mapsto \Psi_\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) \tag{2}$$

from $\mathcal{T}$ into $\mathcal{T}$ such that the following conditions are satisfied:

(a) the mapping (2) can be computed in $\mathcal{O}(f)$-time,
(b) if $\varphi(\boldsymbol{y}) \wedge (\exists \boldsymbol{z})\Psi_\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is satisfiable, so is $\varphi(\boldsymbol{y}) \wedge \psi(\boldsymbol{x})$,
(c) $\models \big(\varphi(\boldsymbol{y}) \ \wedge\ \psi(\boldsymbol{x})\big) \ \longrightarrow\ (\exists \boldsymbol{z})\Psi_\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.

It turns out that standard expressibility is a special case of $\mathcal{O}(1)$-expressibility. This is stated in the following lemma, whose simple proof is provided in Appendix A.

Lemma 1. *If a formula $\psi(\boldsymbol{x})$ is expressible in a fragment $\mathcal{T}$ of $\mathbb{BST}$, then it is also $\mathcal{O}(1)$-expressible in $\mathcal{T}$.*

Various expressibility and inexpressibility results are collected in the following lemma proved in Appendix A.

Lemma 2. *(a) $x = y \setminus z$ is expressible in $\mathsf{BST}(\cup, \mathsf{Disj}, =)$;*
(b) $x = y \cap z$ and $x = y \cup z$ are expressible in $\mathsf{BST}(\setminus, =)$;
(c) $x = y$ is expressible in $\mathsf{BST}(\subseteq)$;
(d) $x \subseteq y$ is expressible both in $\mathsf{BST}(\cup, =)$ and in $\mathsf{BST}(\cap, =)$;
(e) $x \not\subseteq y$ is expressible both in $\mathsf{BST}(\cup, \neq)$ and in $\mathsf{BST}(\cap, \neq)$;
(f) $x \neq \varnothing$ is expressible in $\mathsf{BST}(\subseteq, \neq)$, and therefore in $\mathsf{BST}(\cup, =, \neq)$; moreover, $x \neq \varnothing$ is expressible in $\mathsf{BST}(\not\subseteq)$, in $\mathsf{BST}(\neq, \mathsf{Disj})$, in $\mathsf{BST}(=\!\varnothing, \neq)$, and in $\mathsf{BST}(\neg\mathsf{Disj})$;
(g) $x = \varnothing$ is expressible in $\mathsf{BST}(\mathsf{Disj})$;
(h) $\mathsf{Disj}(x, y)$ is expressible both in $\mathsf{BST}(\cap, =\!\varnothing)$ and in $\mathsf{BST}(\setminus, =)$, and $\neg\mathsf{Disj}(x, y)$ is expressible both in $\mathsf{BST}(\cap, \neq\!\varnothing)$ and in $\mathsf{BST}(\subseteq, \neq\!\varnothing)$;
(i) $\neg\mathsf{Disj}(x, y)$ (i.e., $x \cap y \neq \varnothing$) is expressible in $\mathsf{BST}(\subseteq, \neq)$, and therefore expressible in $\mathsf{BST}(\cup, =, \neq)$;
(j) $x = \varnothing$ is not expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$;
(k) $x = y \setminus z$ is not expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$.

The following lemma, whose proof can be found in Appendix A, allows us to infer that the literal $x = \varnothing$ is $\mathcal{O}(n)$-expressible in $\mathsf{BST}(\cup, =, \neq)$.

Lemma 3. *The mapping $\varphi(\boldsymbol{y}) \mapsto \Psi_\varphi(\boldsymbol{y}, x)$ from $\mathsf{BST}(\cup, =, \neq)$ into itself, where x is any set variable (possibly in φ) and*

$$\Psi_\varphi(\boldsymbol{y}, x) \coloneqq \bigwedge_{z \in \mathit{Vars}(\varphi)} z \cup x = z,$$

enjoys the properties

- *if $\varphi(\boldsymbol{y}) \wedge \Psi_\varphi(\boldsymbol{y}, x)$ is satisfiable, so is $\varphi(\boldsymbol{y}) \wedge x = \varnothing$, and*
- *$\models \big(\varphi(\boldsymbol{y}) \ \wedge\ x = \varnothing\big) \ \longrightarrow\ \Psi_\varphi(\boldsymbol{y}, x)$.*

Hence, the literal $x = \varnothing$ is $\mathcal{O}(n)$-expressible by $\Psi_\varphi(\boldsymbol{y}, x)$ in $\mathsf{BST}(\cup, =, \neq)$.

2 Complexity taxonomy of the fragments of $\mathbb{BST}$

Of a fragment of $\mathbb{BST}$, we say that *it is* NP*-complete* if it has an NP-complete satisfiability problem (see [11]). Likewise, we say that *it is polynomial* if its satisfiability problem has polynomial complexity.

The number of distinct fragments of $\mathbb{BST}$ is equal to $2^3 \cdot (2^8 - 1) = 2040$, of which 1278 are NP-complete and the remaining 762 are polynomial. The complexity of any fragment of $\mathbb{BST}$ can be efficiently identified once the *minimal* NP-complete fragments (namely the NP-complete fragments of $\mathbb{BST}$ that do not strictly contain any NP-complete fragment of $\mathbb{BST}$) and the *maximal* polynomial fragments (namely the polynomial fragments of $\mathbb{BST}$ that are not strictly contained in any polynomial fragment of $\mathbb{BST}$) have been singled out. Indeed, any $\mathbb{BST}$-fragment either is contained in some maximal polynomial $\mathbb{BST}$-fragment or contains some minimal NP-complete fragment.

$\cup$	$\cap$	$\setminus$	$=\varnothing$	$\neq\varnothing$	Disj	$\neg$Disj	$\subseteq$	$\not\subseteq$	$=$	$\neq$	Complexity
		$\star$								$\star$	NP-complete
		$\star$						$\star$			NP-complete
		$\star$				$\star$					NP-complete
		$\star$		$\star$							NP-complete
$\star$	$\star$									$\star$	NP-complete
$\star$	$\star$							$\star$			NP-complete
$\star$	$\star$		$\star$	$\star$							NP-complete
$\star$	$\star$				$\star$	$\star$					NP-complete
$\star$	$\star$		$\star$			$\star$					NP-complete
$\star$	$\star$			$\star$	$\star$						NP-complete
$\star$					$\star$	$\star$			$\star$		NP-complete
$\star$				$\star$	$\star$				$\star$		NP-complete
$\star$					$\star$	$\star$	$\star$				NP-complete
$\star$					$\star$				$\star$	$\star$	NP-complete
$\star$					$\star$			$\star$	$\star$		NP-complete
$\star$				$\star$	$\star$		$\star$				NP-complete
$\star$					$\star$		$\star$			$\star$	NP-complete
$\star$					$\star$		$\star$	$\star$			NP-complete
$\star$	$\star$	$\star$	$\star$		$\star$		$\star$		$\star$		$\mathcal{O}(1)$
$\star$	$\star$			$\star$		$\star$	$\star$		$\star$		$\mathcal{O}(1)$
$\star$			$\star$	$\star$	$\star$	$\star$		$\star$		$\star$	$\mathcal{O}(n^3)$
	$\star$		$\star$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$	$\mathcal{O}(n^4)$
$\star$			$\star$	$\star$		$\star$	$\star$	$\star$	$\star$	$\star$	$\mathcal{O}(n^3)$

Table 1. Complete taxonomy of minimal NP-complete and maximal polynomial fragments of $\mathbb{BST}$

Table 1 reports the 18 minimal NP-complete fragments of $\mathbb{BST}$ and the 5 maximal polynomial fragments of $\mathbb{BST}$. Each row represents the fragment involving the operators and the relators that are marked with a '$\star$' symbol.

2.1 Minimal NP-complete fragments of $\mathbb{BST}$

Concerning the NP-complete fragments, initially we proved that the fragments

$$\mathsf{BST}(\setminus, \neq), \quad \mathsf{BST}(\cup, \cap, \neq), \quad \mathsf{BST}(\cup, \cap, =\varnothing, \neq\varnothing), \text{ and } \quad \mathsf{BST}(\cup, =, \mathsf{Disj}, \neg\mathsf{Disj})$$

are NP-complete, by reducing the well-known NP-complete problem $\mathsf{3SAT}$ (see [11]) to each of them.

Then it can be observed that:

first block: the NP-completeness of the fragments $\mathsf{BST}(\setminus, \not\subseteq)$, $\mathsf{BST}(\setminus, \neg\mathsf{Disj})$, and $\mathsf{BST}(\setminus, \neq\varnothing)$ in the first block of Table 1 can be obtained by much the same technique used to reduce $\mathsf{3SAT}$ to $\mathsf{BST}(\setminus, \neq)$;

second block: the NP-completeness of the fragment $\mathsf{BST}(\cup, \cap, \not\subseteq)$ can be achieved by much the same technique used to reduce $\mathsf{3SAT}$ to $\mathsf{BST}(\cup, \cap, \neq)$;

third block: the NP-completeness of the fragments $\mathsf{BST}(\cup, \cap, \mathsf{Disj}, \neg\mathsf{Disj})$, $\mathsf{BST}(\cup, \cap, =\varnothing, \neg\mathsf{Disj})$, and $\mathsf{BST}(\cup, \cap, \neq\varnothing, \mathsf{Disj})$ can be obtained by much the same technique used to reduce $\mathsf{3SAT}$ to $\mathsf{BST}(\cup, \cap, =\varnothing, \neq\varnothing)$; and

fourth block: the NP-completeness of the fragment $\mathsf{BST}(\cup, =, \neq\varnothing, \mathsf{Disj})$ can be shown by much the same reduction technique used for $\mathsf{BST}(\cup, =, \mathsf{Disj}, \neg\mathsf{Disj})$.

Finally, by resorting to some of the expressibility results listed in Lemma 2, it can readily be proved that:

- $\mathsf{BST}(\cup, =, \mathsf{Disj}, \neg\mathsf{Disj})$ can be reduced in linear time
  to $\mathsf{BST}(\cup, \subseteq, \mathsf{Disj}, \neg\mathsf{Disj})$, by Lemma 2(c),
- $\mathsf{BST}(\cup, =, \neq\varnothing, \mathsf{Disj})$ can be reduced in linear time
  to $\mathsf{BST}(\cup, =, \neq, \mathsf{Disj})$, by Lemma 2(f),
  to $\mathsf{BST}(\cup, =, \not\subseteq, \mathsf{Disj})$, by Lemma 2(f),
  to $\mathsf{BST}(\cup, \subseteq, \neq\varnothing, \mathsf{Disj})$, by Lemma 2(c),
  to $\mathsf{BST}(\cup, \subseteq, \neq, \mathsf{Disj})$, by Lemma 2(c)(f),
  to $\mathsf{BST}(\cup, \subseteq, \not\subseteq, \mathsf{Disj})$, by Lemma 2(c)(f).

2.2 Maximal polynomial fragments of $\mathbb{BST}$

Two of the maximal polynomial fragments of $\mathbb{BST}$, namely

$$\mathsf{BST}(\cup, \cap, \setminus, =\varnothing, \mathsf{Disj}, \subseteq, =) \qquad \text{and} \qquad \mathsf{BST}(\cup, \cap, =, \neq\varnothing, \neg\mathsf{Disj}, \subseteq)$$

are trivial as they contain only satisfiable conjunctions, thereby admitting a $\mathcal{O}(1)$ satisfiability test.

Notice that the first fragment comprises all the positive relators and the complete suite of Boolean operators. It is immediate to check that each of its conjunctions φ is satisfied by the *null* set assignment $M_\emptyset$ over $\mathit{Vars}(\varphi)$, where $M_\emptyset x = \emptyset$ for each $x \in \mathit{Vars}(\varphi)$.

Concerning the second fragment, it can easily be verified that each of its conjunctions ψ is satisfied by any *constant nonnull* set assignment M_a over $\mathit{Vars}(\psi)$, where a is a nonempty set and $M_a x = a$ for every $x \in \mathit{Vars}(\psi)$.

Next, we provided $\mathcal{O}(n^3)$ satisfiability tests for the fragments $\mathsf{BST}(\cup, \mathsf{Disj}, \neg\mathsf{Disj}, \neq)$ and $\mathsf{BST}(\cup, =, \neq)$, and a $\mathcal{O}(n^4)$ satisfiability test for the fragment $\mathsf{BST}(\cap, =\varnothing, =, \neq)$.

Since

- $\models x \nsubseteq y \longleftrightarrow x \cup y \neq y$ (cf. Lemma 2(e)) and
- $\models x = \varnothing \longleftrightarrow \mathsf{Disj}(x, x)$ (cf. Lemma 2(f),(g)),

the $\mathcal{O}(n^3)$ satisfiability test for $\mathsf{BST}(\cup, \mathsf{Disj}, \neg\mathsf{Disj}, \neq)$ yields a $\mathcal{O}(n^3)$ satisfiability test for $\mathsf{BST}(\cup, =\varnothing, \neq\varnothing, \mathsf{Disj}, \neg\mathsf{Disj}, \nsubseteq, \neq)$.

In addition, since

- $x \neq \varnothing$ is expressible in $\mathsf{BST}(=\varnothing, \neq)$ (cf. Lemma 2(f)),
- $\mathsf{Disj}(x, y)$ and $\neg\mathsf{Disj}(x, y)$ are expressible in $\mathsf{BST}(\cap, = \varnothing, \neq \varnothing)$ (cf. Lemma 2(h)), and
- $x \subseteq y$ and $x \nsubseteq y$ are expressible in $\mathsf{BST}(\cap, =\varnothing, \neq\varnothing)$ (cf. Lemma 2(d),(e)),

it follows that the $\mathcal{O}(n^4)$ satisfiability test for $\mathsf{BST}(\cap, =\varnothing, =, \neq)$ yields a $\mathcal{O}(n^4)$ satisfiability test for the fragment $\mathsf{BST}(\cap, =\varnothing, \neq\varnothing, \mathsf{Disj}, \neg\mathsf{Disj}, \subseteq, \nsubseteq, =, \neq)$.

Finally, since

- $x = \varnothing$ is $\mathcal{O}(n)$-expressible in $\mathsf{BST}(\cup, =, \neq)$ (cf. Lemma 3),
- $x \neq \varnothing$ is expressible in $\mathsf{BST}(=\varnothing, \neq)$ (cf. Lemma 2(f)),
- $x \subseteq y$ is expressible in $\mathsf{BST}(\cup, =)$,
- $x \nsubseteq y$ is expressible in $\mathsf{BST}(\cup, \neq)$, and
- $\neg\mathsf{Disj}(x, y)$ is expressible in $\mathsf{BST}(\neq\varnothing, \subseteq)$,

the $\mathcal{O}(n^3)$ satisfiability test for $\mathsf{BST}(\cup, =, \neq)$ yields a $\mathcal{O}(n^3)$ satisfiability test for the fragment $\mathsf{BST}(\cup, =\varnothing, \neq\varnothing, \neg\mathsf{Disj}, \subseteq, \nsubseteq, =, \neq)$.

It can be checked that:

(A) none of the fragments listed in Table 1 is strictly contained in another fragment in the same table, and
(B) for every fragment $\mathcal{T}$ of $\mathbb{BST}$, there is a fragment in Table 1 that either contains $\mathcal{T}$ or is contained in $\mathcal{T}$.

Properties (A) and (B) imply that the 18 NP-complete fragments in Table 1 are indeed minimally NP-complete and, symmetrically, the 5 polynomial fragments in Table 1 are maximally polynomial.

$\cup$	$\cap$	$\setminus$	$=\varnothing$	$\neq\varnothing$	Disj	$\neg\mathsf{Disj}$	$\subseteq$	$\not\subseteq$	$=$	$\neq$	Complexity
$\star$			$\star$	$\star$	$\star$			$\star$		$\star$	$\mathcal{O}(n)$
	$\star$		$\star$	$\star$	$\star$					$\star$	$\mathcal{O}(n^3)$

Table 2. Two non-maximal polynomial fragments of $\mathbb{BST}$

2.3 A linear satisfiability test for $\mathbb{BST}(\cup, \mathsf{Disj}, \neq)$

While there is a limited interest in further investigating the non-minimal NP-complete fragments of $\mathbb{BST}$, this is not the case for the non-maximal polynomial fragments, as the latter can admit decision tests more efficient than any of the maximal polynomial fragments extending them.

We briefly report here some preliminary results obtained so far in this direction (see Table 2). In particular, we devised:

- a linear-time decision test for the fragment $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$, which readily generalizes, by Lemma 2(e),(f),(g), to a linear-time satisfiability test for the extended fragment $\mathsf{BST}(\cup, =\varnothing, \neq\varnothing, \mathsf{Disj}, \not\subseteq, \neq)$;
- a cubic algorithm for the fragment $\mathsf{BST}(\cap, =\varnothing, \neq)$, which yields, by Lemma 2(f),(h), a cubic satisfiability test for the extended fragment $\mathsf{BST}(\cap, =\varnothing, \neq\varnothing, \mathsf{Disj}, \neq)$.

For space reasons, we limit ourselves to present only a linear-time satisfiability test for the fragment $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$.

For convenience, we shall represent terms of the form $x_1 \cup \cdots \cup x_h$ as $\bigcup\{x_1, \ldots, x_h\}$. Thus, for a set assignment M and a finite nonempty collection of set variables $L \subseteq \mathit{Vars}(\varphi)$, we shall have $M(\bigcup L) = \bigcup ML = \bigcup_{x \in L} Mx$.

Towards a linear satisfiability test for $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$, let φ be a satisfiable $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$-conjunction of the form

$$\bigwedge_{i=1}^{p} \bigcup L_i \neq \bigcup R_i \ \wedge \bigwedge_{j=p+1}^{q} \mathsf{Disj}(\bigcup L_j, \bigcup R_j), \tag{3}$$

where the L_h's and the R_h's are nonempty collections of set variables, and let M be a set assignment over $\mathit{Vars}(\varphi)$ satisfying φ.

Preliminarily, we observe that, for each $x \in \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j)$, $Mx \subseteq M(\bigcup L_j) \cap M(\bigcup R_j)$ holds for some $j \in \{p+1, \ldots, q\}$. Hence, since the conjunct $\mathsf{Disj}(\bigcup L_j, \bigcup R_j)$ occurs in φ, we have $M(\bigcup L_j) \cap M(\bigcup R_j) = \emptyset$, which in turn yields $Mx = \emptyset$.

Next, for each conjunct of type $\bigcup L_i \neq \bigcup R_i$ in φ, if any, we have $M(\bigcup L_i) \neq M(\bigcup R_i)$, i.e., $\big(M(\bigcup L_i) \cup M(\bigcup R_i)\big) \setminus \big(M(\bigcup L_i) \cap M(\bigcup R_i)\big) \neq \emptyset$, and *a fortiori* $\big(M(\bigcup L_i) \cup M(\bigcup R_i)\big) \neq \emptyset$. Thus, there is an $x \in \bigcup L_i \cup \bigcup R_i$ such that $Mx \neq \emptyset$, and therefore the preceding observaton implies $(\bigcup L_i \cup \bigcup R_i) \setminus \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j) \neq \emptyset$. Summarizing, we have shown that, by assuming the satisfiability of φ, then the following condition is fulfilled:

(C1) $(\bigcup L_i \cup \bigcup R_i) \setminus \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j) \neq \emptyset$, for every $i = 1, \ldots, p$.

Conversely, let φ be a $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$-conjunction of the form (3) for which the condition (C1) is true, and let $x_1, \ldots, x_k$ be the distinct variables in $\mathit{Vars}(\varphi) \setminus \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j)$. Consider any assignment M^* over $\mathit{Vars}(\varphi)$ such that

- $M^* \coloneqq \emptyset$, for each $x \in \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j)$, and
- $M^*x_1, \ldots, M^*x_k$ are nonempty pairwise disjoint sets.

Then, it is not hard to check that M^* satisfies φ. Thus, we have proved the following lemma, which readily yields a satisfiability test for $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$.

Lemma 4. *Let φ be a $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$-conjunction of the form* (3)*. Then φ is satisfiable if and only if condition (C1) holds.*

Concerning the complexity of the satisfiability test implicit in Lemma 4, we observe that condition (C1) can be tested in $\mathcal{O}(|\varphi|)$ time, since

- the set $\bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j)$ can be computed in $\mathcal{O}\big(\sum_{j=p+1}^{q}(|L_j|+|R_j|)\big) = \mathcal{O}(|\varphi|)$ time;
- the set $(\bigcup L_i \cup \bigcup R_i) \setminus \bigcup_{j=p+1}^{q}(\bigcup L_j \cap \bigcup R_j)$ can be computed in $\mathcal{O}(|L_i|+|R_i|)$ time and tested for emptiness in constant time, for each $i = 1m, \ldots, p$, for an overall $\mathcal{O}\big(\sum_{i=1}^{p}(|L_i| + |R_i| + 1)\big) = \mathcal{O}(|\varphi|)$ time.

Hence, we have:

Lemma 5. *The satisfiability problem for $\mathsf{BST}(\cup, \mathsf{Disj}, \neq)$-conjunctions can be solved in linear time.*

3 Conclusion and future work

We highlighted some preliminary results of an investigation aimed at identifying small fragments of set theory endowed with polynomial-time satisfiability decision tests, potentially useful for automated proof verification. In this initial phase, we mainly focused on 'Boolean Set Theory', namely the fragment of quantifier-free formulae of set theory involving variables, the Boolean set operators $\cup, \cap, \setminus$, the Boolean relators $\subseteq, \not\subseteq, =, \neq$, and the predicates '$\cdot = \varnothing$' and '$\mathsf{Disj}(\cdot, \cdot)$', along with their opposites.

Future work will concentrate on the analysis of the sub-maximal polynomial fragments of $\mathbb{BST}$, so as to obtain a finer complexity taxonomy of the collection of all $\mathbb{BST}$ fragments, and also on enriching the endowment of set operators and relators of $\mathbb{BST}$. We also plan to further deepen the study of membership fragments of the types presented in Appendix B and, ultimately, to explore the cases in which relators correlated to membership and Boolean set predicates are allowed simultaneously.

References

1. Domenico Cantone. Decision procedures for elementary sublanguages of set theory. X. Multilevel syllogistic extended by the singleton and powerset operators. *J. Automat. Reason.*, 7(2):193–230, June 1991.

2. Domenico Cantone, Vincenzo Cutello, and Alberto Policriti. Set-theoretic reductions of Hilbert's tenth problem. In Egon Börger, Hans Kleine Büning, and Michael M. Richter, editors, *CSL '89, 3rd Workshop on Computer Science Logic, Kaiserslautern, Germany, October 2-6, 1989, Proceedings*, volume 440 of *Lecture Notes in Computer Science*, pages 65–75. Springer, 1990.
3. Domenico Cantone, Alfredo Ferro, and Eugenio G. Omodeo. *Computable set theory.* Number 6 in International Series of Monographs on Computer Science, Oxford Science Publications. Clarendon Press, Oxford, UK, 1989.
4. Domenico Cantone, Eugenio G. Omodeo, and Alberto Policriti. The automation of syllogistic. II: Optimization and complexity issues. *J. Automat. Reason.*, 6(2):173–187, June 1990.
5. Domenico Cantone, Eugenio G. Omodeo, and Alberto Policriti. *Set theory for computing - From decision procedures to declarative programming with sets.* Monographs in Computer Science. Springer-Verlag, New York, 2001.
6. Domenico Cantone, Eugenio G. Omodeo, and Pietro Ursino. Formative processes with applications to the decision problem in set theory: I. Powerset and singleton operators. *Information and Computation*, 172(2):165–201, January 2002.
7. Domenico Cantone and Pietro Ursino. Formative processes with applications to the decision problem in set theory: II. Powerset and singleton operators, finiteness predicate. *Information and Computation*, 237:215–242, 2014.
8. Domenico Cantone and Pietro Ursino. *An Introduction to the Technique of Formative Processes in Set Theory.* Springer International Publishing, 2018.
9. Domenico Cantone and Calogero G. Zarba. A new fast tableau-based decision procedure for an unquantified fragment of set theory. In R. Caferra and G. Salzer, editors, *Automated Deduction in Classical and Non-Classical Logics*, volume 1761 of *Lecture Notes in Artificial Intelligence*, pages 127–137. Springer-Verlag, 2000.
10. Alfredo Ferro, Eugenio G. Omodeo, and Jacob T. Schwartz. Decision procedures for elementary sublanguages of set theory. I: Multilevel syllogistic and some extensions. *Commun. Pur. Appl. Math.*, 33:599–608, 1980.
11. Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Series of Books in the Mathematical Sciences. W. H. Freeman, 1979.
12. Nathan Jacobson. *Lectures in Abstract Algebra, Vol.1 – Basic Concepts.* D. Van Nostrand, New York, 1951.
13. Yuri Ivanovich Manin. *A course in mathematical logic.* Graduate texts in Mathematics. Springer-Verlag, 1977.
14. E. G. Omodeo and A. I. Tomescu. Set graphs. III. Proof Pearl: Claw-free graphs mirrored into transitive hereditarily finite sets. *J. Automat. Reason.*, 52(1):1–29, 2014.
15. Eugenio G. Omodeo, Alberto Policriti, and Alexandru I. Tomescu. *On Sets and Graphs: Perspectives on Logic and Combinatorics.* Springer, 2017.
16. Jacob T. Schwartz, Domenico Cantone, and Eugenio G. Omodeo. *Computational logic and set theory: Applying formalized logic to analysis.* Springer-Verlag, 2011. Foreword by M. Davis.

A Proofs of Lemmas 1, 2, and 3

Proof of Lemma 1. Let $\psi(\boldsymbol{x})$ be any formula expressible in $\mathcal{T}$, and let $\Psi(\boldsymbol{x}, \boldsymbol{z})$ be a $\mathcal{T}$-conjunction such that

$$\models \psi(\boldsymbol{x}) \longleftrightarrow (\exists \boldsymbol{z})\Psi(\boldsymbol{x}, \boldsymbol{z})\,. \tag{4}$$

Consider the mapping

$$\varphi(\boldsymbol{y}) \mapsto \Psi(\boldsymbol{x}, \boldsymbol{z}) \tag{5}$$

from $\mathcal{T}$ into $\mathcal{T}$, where $\boldsymbol{z}$ is any tuple of distinct set variables. Plainly, the mapping (5) can be computed in $\mathcal{O}(1)$-time. In addition, by (4), we have

$$\models \big(\varphi(\boldsymbol{y}) \wedge \psi(\boldsymbol{x})\big) \;\longleftrightarrow\; \big(\varphi(\boldsymbol{y}) \wedge (\exists \boldsymbol{z})\Psi(\boldsymbol{x}, \boldsymbol{z})\big).$$

Hence, in particular, the formulae $\varphi(\boldsymbol{y}) \wedge \psi(\boldsymbol{x})$ and $\varphi(\boldsymbol{y}) \wedge (\exists \boldsymbol{z})\Psi(\boldsymbol{x}, \boldsymbol{z})$ are equi-satisfiable, and we also have

$$\models \big(\varphi(\boldsymbol{y}) \wedge \psi(\boldsymbol{x})\big) \;\longrightarrow\; (\exists \boldsymbol{z})\Psi(\boldsymbol{x}, \boldsymbol{z}).$$

Thus, conditions (b) and (c) of Definition 2 are also satisfied, proving that the formula $\psi(\boldsymbol{x})$ is $\mathcal{O}(1)$-expressible in $\mathcal{T}$. $\square$

Proof of Lemma 2. (a) $\models x = y \setminus z \;\longleftrightarrow\; (x \cap z = \varnothing \;\wedge\; x \cup z = y \cup z)$.

(b) $\models x = y \cup z \;\longleftrightarrow\; \big(x \setminus y = z \setminus y \;\wedge\; x \setminus (z \setminus y) = y\big)$ and
$\models x = y \cap z \;\longleftrightarrow\; x = y \setminus (y \setminus z)$.

(c) $\models x = y \longleftrightarrow x \subseteq y \;\wedge y \subseteq x$.

(d) $\models x \subseteq y \longleftrightarrow x \cup y = y$ and $\models x \subseteq y \longleftrightarrow x \cap y = x$

(e) $\models x \nsubseteq y \longleftrightarrow x \cup y \neq y$ and $\models x \nsubseteq y \longleftrightarrow x \cap y \neq x$

(f) We have that:
- $\models x \neq \varnothing \;\longleftrightarrow\; (\exists y, z)(y \subseteq x \;\wedge\; z \subseteq x \;\wedge\; z \neq y)$,
- from the latter and (c), it follows that $x \neq \varnothing$ is also expressible in $\mathsf{BST}(\cup, =, \neq)$,
- $\models x \neq \varnothing \;\longleftrightarrow\; (\exists y)(x \nsubseteq y)$,
- $\models x \neq \varnothing \;\longleftrightarrow\; (\exists y)(x \neq y \;\wedge\; \mathsf{Disj}(y, y))$,
- $\models x \neq \varnothing \;\longleftrightarrow\; (\exists y)(x \neq y \;\wedge\; y = \varnothing)$,
- $\models x \neq \varnothing \;\longleftrightarrow\; \neg\mathsf{Disj}(x, x)$,

(g) $\models x = \varnothing \longleftrightarrow \mathsf{Disj}(x, x)$.

(h) We have that:
- $\models \mathsf{Disj}(x, y) \longleftrightarrow x \cap y = \varnothing$ and $\models \mathsf{Disj}(x, y) \longleftrightarrow x \setminus (x \setminus y) = x \setminus x$,
- $\models \neg\mathsf{Disj}(x, y) \longleftrightarrow x \cap y \neq \varnothing$ and $\models \neg\mathsf{Disj}(x, y) \longleftrightarrow (\exists z)(z \subseteq x \;\wedge\; z \subseteq y \;\wedge\; z \neq \varnothing)$.

(i) $\models x \cap y \neq \varnothing \;\longleftrightarrow\; (\exists w, w')\big(w \subseteq x \;\wedge\; w \subseteq y \;\wedge\; w' \subseteq w \;\wedge\; w' \neq w\big)$.
From the latter and (d), it follows that $\neg\mathsf{Disj}(x, y)$ is also expressible in $\mathsf{BST}(\cup, =, \neq)$.

(j) By way of contradiction, assume that there exists a $\mathsf{BST}(\cup, \cap, =, \neq)$-conjunction $\Phi_\varnothing(x, \boldsymbol{y})$ such that

$$\models x = \varnothing \;\longleftrightarrow\; (\exists \boldsymbol{y})\; \Phi_\varnothing(x, \boldsymbol{y}) \tag{6}$$

(so that $(\exists \boldsymbol{y})\Phi_\varnothing(x, \boldsymbol{y})$, and therefore $\Phi_\varnothing(x, \boldsymbol{y})$, is satisfiable).
Let M be any set assignment such that $M \models \Phi_\varnothing(x, \boldsymbol{y})$, and set $M'z \coloneqq Mz \cup C$, for every $z \in \mathit{Vars}(\Phi_\varnothing)$, where C is any nonempty set that is disjoint from

Mz, for every $z \in \mathit{Vars}(\Phi_\varnothing)$ (namely, such that $C \cap \bigcup M(\mathit{Vars}(\Phi_\varnothing)) = \emptyset$). Then, for $y_1, \ldots, y_n \in \mathit{Vars}(\Phi_\varnothing)$, we have:

$$\begin{aligned} M'(y_1 \cup \ldots \cup y_n) &= M'y_1 \cup \ldots \cup M'y_n \\ &= (My_1 \cup C) \cup \ldots \cup (My_n \cup C) \\ &= (My_1 \cup \ldots \cup My_n) \cup C \\ &= M(y_1 \cup \ldots \cup y_n) \cup C \end{aligned}$$

and

$$\begin{aligned} M'(y_1 \cap \ldots \cap y_n) &= M'y_1 \cap \ldots \cap M'y_n \\ &= (My_1 \cup C) \cap \ldots \cap (My_n \cup C) \\ &= (My_1 \cap \ldots \cap My_n) \cup C \\ &= M(y_1 \cap \ldots \cap y_n) \cup C. \end{aligned}$$

Therefore:

(j_1) if the literal $y_1 \cup \ldots \cup y_n = z_1 \cup \ldots \cup z_m$ is in $\Phi_\varnothing$, then

$$\begin{aligned} M'(y_1 \cup \ldots \cup y_n) &= M(y_1 \cup \ldots \cup y_n) \cup C \\ &= M(z_1 \cup \ldots \cup z_m) \cup C = M'(z_1 \cup \ldots \cup z_m); \end{aligned}$$

(j_2) if the literal $y_1 \cap \ldots \cap y_n = z_1 \cap \ldots \cap z_m$ is in $\Phi_\varnothing$, then

$$\begin{aligned} M'(y_1 \cap \ldots \cap y_n) &= M(y_1 \cap \ldots \cap y_n) \cup C \\ &= M(z_1 \cap \ldots \cap z_m) \cup C = M'(z_1 \cap \ldots \cap z_m); \end{aligned}$$

(j_3) if the literal $y \neq z$ is in $\Phi_\varnothing$, then we have

$$M'y = My \cup C, \quad M'z = Mz \cup C, \quad \text{and} \quad My \neq Mz.$$

Since C is disjoint from My and Mz, then $M'y \neq M'z$.

From (j_1), (j_2), and (j_3), it follows that $M' \models \Phi_\varnothing$, so that we have also $M' \models (\exists \boldsymbol{y})\Phi_\varnothing$.

In addition, we have $M'x = Mx \cup C \neq \emptyset$. Thus, $M' \not\models (\exists \boldsymbol{y})\Phi_\varnothing \longrightarrow x = \varnothing$, contradicting (6), hence showing that $x = \varnothing$ is not expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$.

(k) If $x = y \setminus z$ were expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$, then there would exist a conjunction $\Psi(x, y, z, \boldsymbol{w})$ in $\mathsf{BST}(\cup, \cap, =, \neq)$ such that

$$\models x = y \setminus z \longleftrightarrow (\exists \boldsymbol{w})\Psi(x, y, z, \boldsymbol{w}). \tag{7}$$

From (7), we have

$$\models x = y \setminus y \longleftrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}).$$

Since

$$
\begin{aligned}
&\models x = y \setminus y \longleftrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}) \\
&\quad \Longrightarrow \models x = \varnothing \longleftrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}) \\
&\quad \Longrightarrow \models (\forall y)\big[x = \varnothing \longleftrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w})\big] \\
&\quad \Longrightarrow \models (\forall y)\big[\big((\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}) \longrightarrow x = \varnothing\big) \\
&\qquad\qquad\qquad\qquad \wedge \big(x = \varnothing \longrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w})\big)\big] \\
&\quad \Longrightarrow \models \big[(\forall y)((\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}) \longrightarrow x = \varnothing) \\
&\qquad\qquad\qquad\qquad \wedge (\forall y)\big(x = \varnothing \longrightarrow (\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w})\big)\big] \\
&\quad \Longrightarrow \models \big[\big((\forall y)\neg(\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w}) \vee x = \varnothing\big) \\
&\qquad\qquad\qquad\qquad \wedge \big(x = \varnothing \longrightarrow (\forall y)(\exists \boldsymbol{w})\Psi(x, y, y, \boldsymbol{w})\big)\big] \\
&\quad \Longrightarrow \models \big[\big(\neg(\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w}) \vee x = \varnothing\big) \\
&\qquad\qquad\qquad\qquad \wedge \big(x = \varnothing \longrightarrow (\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w})\big)\big] \\
&\quad \Longrightarrow \models \big[\big((\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w}) \longrightarrow x = \varnothing\big) \\
&\qquad\qquad\qquad\qquad \wedge \big(x = \varnothing \longrightarrow (\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w})\big)\big] \\
&\quad \Longrightarrow \models x = \varnothing \longleftrightarrow (\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w}),
\end{aligned}
$$

then

$$\models x = \varnothing \longleftrightarrow (\exists \boldsymbol{w})(\exists y)\Psi(x, y, y, \boldsymbol{w}),$$

i.e., $x = \varnothing$ would be expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$, contradicting (j). Thus, $x = y \setminus z$ is not expressible in $\mathsf{BST}(\cup, \cap, =, \neq)$. $\square$

We state without proof the following model-theoretic property for the fragment $\mathsf{BST}(\cup, =, \neq)$.

Proposition 1. *Let φ be a satisfiable conjunction of $\mathsf{BST}(\cup, =, \neq)$ and x any variable such that*

$$\models \varphi \wedge \bigwedge_{z \in Vars(\varphi)} z \cup x = z.$$

Then φ has a model M such that $Mx = \emptyset$.

Proof of Lemma 3. Let $\varphi \mapsto \Psi_\varphi(\boldsymbol{y}, x)$ be the mapping from $\mathsf{BST}(\cup, =, \neq)$ into itself where

$$\Psi_\varphi(\boldsymbol{y}, x) := \bigwedge_{z \in Vars(\varphi)} z \cup x = z.$$

Plainly, $\models x = \varnothing \longrightarrow \bigwedge_{z \in Vars(\varphi)} z \cup x = z$. Hence, *a fortiori*,

$$\models (\varphi(\boldsymbol{y}) \ \wedge \ x = \varnothing) \longrightarrow \bigwedge_{z \in Vars(\varphi)} z \cup x = z,$$

proving condition (c) of Definition 2.

As for condition (b) of Definition 2, we have to show that if $\varphi(\boldsymbol{y}) \wedge \bigwedge_{z \in Vars(\varphi)} z \cup x = z$ is satisfiable, so is $\varphi(\boldsymbol{y}) \wedge x = \varnothing$. But this follows at once from Proposition 1.

Finally, by observing that the mapping $\varphi \mapsto \bigwedge_{z \in Vars(\varphi)} z \cup x = z$ can clearly be computed in $\mathcal{O}(n)$-time, it follows that the literal $x = \varnothing$ can be $\mathcal{O}(n)$-expressed in $\mathsf{BST}(\cup, =, \neq)$. $\square$

B Complexity taxonomy for small fragments in presence of the membership relation

In this appendix we present a quick overview of the complexity taxonomy for the fragment $\mathsf{Th}(\cup, \cap, \setminus, \in, \notin)$, which from now on we refer to as $\mathbb{MST}$, acronym for *membership set theory*.

The number of distinct fragments of $\mathbb{MST}$ is 24, of which 10 are NP-complete and 14 are polynomial. Much as for $\mathbb{BST}$, the complexity of any fragment of $\mathbb{MST}$ can be efficiently determined once the *minimal* NP-complete fragments and the *maximal* polynomial fragments have been singled out. Table 3 shows the minimal NP-complete and the maximal polynomial fragments of $\mathbb{MST}$.

$\cup$	$\cap$	$\setminus$	$\in$	$\notin$	Complexity
		$\star$	$\star$		NP-complete
$\star$	$\star$		$\star$		NP-complete
$\star$			$\star$	$\star$	$\mathcal{O}(n)$
	$\star$		$\star$	$\star$	$\mathcal{O}(n^2)$
$\star$	$\star$	$\star$		$\star$	$\mathcal{O}(1)$

Table 3. Maximal polynomial and minimal NP-complete fragments of $\mathbb{MST}$

The maximal polynomial fragment $\mathsf{MST}(\cup, \cap, \setminus, \notin)$ is trivial as it contains only satisfiable conjunctions, thereby admitting a $\mathcal{O}(1)$ satisfiability test.

Indeed, every $\mathsf{MST}(\cup, \cap, \setminus, \notin)$-conjunction φ is satisfied by the *null* set assignment $M_\emptyset$, sending each $x \in Vars(\varphi)$ to $\emptyset$.

We provided a $\mathcal{O}(n)$ satisfiability test for the fragment $\mathsf{MST}(\cup, \in)$. Then we managed to translate any $\mathsf{MST}(\cup, \in, \notin)$-conjunction into an equisatisfiable $\mathsf{MST}(\cup, \in)$-conjunction in $\mathcal{O}(n)$ time, which resulted in a $\mathcal{O}(n)$ satisfiability test for $\mathsf{MST}(\cup, \in, \notin)$.

We also provided a $\mathcal{O}(n^2)$-time satisfiability test for the fragment $\mathsf{MST}(\cap, \in)$, which was extended into a satisfiability test for the fragment $\mathsf{MST}(\cap, \in, \notin)$ that retains the same complexity.

Finally, the fragments $\mathsf{MST}(\cup, \cap, \in)$ and $\mathsf{MST}(\setminus, \in)$ were proved to be NP-complete. We remark that using the axiom of regularity was necessary to prove the NP-completeness of the former fragment, while regularity was not needed to prove the NP-completeness of the latter.