First-Order Concatenation Theory with Bounded Quantifiers

Lars Kristiansen\textsuperscript{1,2} and Juvenal Murwanashyaka\textsuperscript{1}

\textsuperscript{1} Department of Mathematics, University of Oslo, Norway
\textsuperscript{2} Department of Informatics, University of Oslo, Norway

Abstract. We study first-order concatenation theory with bounded quantifiers. We give axiomatizations with interesting properties, and we prove some normal-form results. Finally, we prove a number of decidability and undecidability results.

1 Introduction

1.1 Concatenation Theory vs. Number Theory

First-order concatenation theory can be compared to first-order number theory, e.g., Peano Arithmetic or Robinson Arithmetic. The universe of a standard structure for first-order number theory is the set of natural numbers. The universe of a standard structure for first-order concatenation theory is the set of finite strings over some alphabet. A first-order language for number theory normally contains two binary functions symbols. In a standard structure these symbols will be interpreted as addition and multiplication. A first-order language for concatenation theory normally contains just one binary function symbol. In a standard structure this symbol will be interpreted as the operator that concatenates two strings. A classical first-order language for concatenation theory—like e.g. the ones studied in Quine \cite{19} and Grzegorczyk \cite{5}—contains no other non-logical symbols apart from constant symbols.

We will stick to a version of concatenation theory where we have a binary alphabet consisting of the bits zero and one, and we will refer to this version as \textit{bit theory}. It is convenient to introduce and explain some notation before we continue our discussion.

1.2 Notation and Terminology

We will use \textbf{0} and \textbf{1} to denote respectively the bits zero and one, and we use pretty standard notation when we work with bit strings: \{\textbf{0}, \textbf{1}\}\* denotes the set of all finite bit strings; \(|\alpha|\) denotes the length of the bit string \(\alpha\); \((\alpha)_i\) denotes the \(i^{th}\) bit of the bit string \(\alpha\); and \textbf{01}\textsuperscript{3}\textbf{0}\textsuperscript{2}\textbf{1} denotes the bit string \textbf{011001}. The set \{\textbf{0}, \textbf{1}\}\* contains the empty string which we will denote \(\varepsilon\).
The language $\mathcal{L}_{BT}$ of first-order bit theory consists of the constants symbols $e, 0, 1$ and the binary function symbol $\circ$. We will use $\mathfrak{B}^-$ to denote the standard $\mathcal{L}_{BT}$-structure: The universe of $\mathfrak{B}^-$ is the set $\{0, 1\}^*$. The constant symbol 0 is interpreted as the string containing nothing but the bit 0, and the constant symbol 1 is interpreted as the string containing nothing but the bit 1, that is, $0^- = 0$ and $1^- = 1$. The constant symbol $e$ is interpreted as the empty string, that is, $e^- = \varepsilon$. Finally, $\circ^-\mathfrak{B}$ is the function that concatenates two strings (e.g. $01^-\circ^- 000^- = 01000^-$ and $\varepsilon^-\circ^- \varepsilon^- = \varepsilon^-$).

We will soon need the biterals. For each $\alpha \in \{0, 1\}^*$ we define the biteral $\overline{\alpha}$ by

$$
\varepsilon = e^-\quad \alpha^-0 = \alpha^-\circ^-0\quad \text{and} \quad \alpha^-1 = \alpha^-\circ^-1.
$$

The biterals correspond to the numerals of first-order number theory: they serve as canonical names for the elements in the universe of the standard structure. Note that, e.g., $((e^-\circ^-1)^-\circ^-1)^-\circ^-0$ is a biteral whereas $(e^-\circ^-1)^-\circ^-1^-\circ^-0$ is not.

### 1.3 Decidability and Undecidability

At a first glance the parsimonious language of first-order bit theory does not seem very expressive, but a little bit of thought shows otherwise. Observe that we can encode a sequence $a_1, a_2, \ldots, a_n$ of natural numbers by a the bit string

$$
00101^{a_1+1}001101^{a_2+1}00101^{a_3+1}00 \ldots 001^{a_n+1}00
.$$ 

Furthermore, we can state that a string is a substring of another (the formula $(\exists uv)[u \circ v = y]$ holds in $\mathfrak{B}^-$ if $x$ is a substring of $y$). We can state that a string contain only ones (the formula $\neg(\exists uv)[u \circ 0 = x]$ holds in $\mathfrak{B}^-$ if $x$ contains only ones, and so does the the formula $x \circ 1 = 1 \circ x$). We can state that a bit string does not contain two consecutive occurrences of zeros, and so on. If we proceed along these lines, we can come up with a formula $\phi(x, y, z)$ such that $\phi(\overline{1^x}, \overline{1^y}, \overline{1^z})$ is true in $\mathfrak{B}$ if $\alpha$ encodes a sequence of natural numbers where the $i^{th}$ element is $a$. The reader interested in the details should consult Section 8 of Leary & Kristiansen [15].

First-order bit theory is indeed expressive enough to code and decode sequences of natural numbers, and then it should be no surprise that the following theorem holds. E.g., it is straightforward to prove the theorem by induction over a Kleene-recursive definition of $f$.

**Theorem 1 (Definability of Computable Functions).** For any (partially) computable function $f : \mathbb{N}^n \to \mathbb{N}$ there exists an $\mathcal{L}_{BT}$-formula $\phi(x_1, \ldots, x_n, y)$ such that

$$
f(a_1, \ldots, a_n) = b \iff \mathfrak{B}^- \models \phi(\overline{1^{a_1}}, \ldots, \overline{1^{a_n}}, \overline{1^b})\,.
$$

Moreover, given a definition of the function, we can compute the formula.

This theorem implies that it is undecidable if a $\mathcal{L}_{BT}$-sentence is true in the standard structure $\mathfrak{B}^-$. It is of course also undecidable if a sentence of first-order
number theory is true in its standard structure \( \mathcal{N} \). Indeed, due to the negative solution of Hilbert’s 10th problem (the Davis-Putnam-Robinson-Matiyasevich Theorem), it is even undecidable if a sentence of the form

\[
\exists x_1, \ldots, x_n \left[ s = t \right]
\]

is true in \( \mathcal{N} \). The bit-theoretic version of Hilbert’s 10th problem turned out to have a positive solution. The next theorem is just a reformulation of a result of Makanin [10].

**Theorem 2 (Makanin).** It is decidable if an \( \mathcal{L}_{BT}^- \)-sentence of the form

\[
\exists x_1, \ldots, x_n \left[ s = t \right]
\]

is true in \( \mathcal{B}^- \).

Hence, it is in general undecidable if a sentence is true in \( \mathcal{B}^- \), but it is decidable if a sentence of the form \( \exists x_1, \ldots, x_n [s = t] \) is true in \( \mathcal{B}^- \). This raises the question: where do we find the border between the decidable and the undecidable? For which subclasses of formulas can we, or can we not, decide truth in the standard structure? Such a question yields an obvious motivation for introducing bounded quantifiers similar to those we know from number theory. Once the bounded quantifiers are there, a number of other questions will knock at the door.

### 1.4 Bounded Quantifiers and \( \Sigma \)-formulas

The first-order language \( \mathcal{L}_{BT}^- \) is \( \mathcal{L}_0^- \) extended by a binary relation symbol \( \sqsubseteq \). We introduce the *bounded existential quantifier* \( (\exists x \sqsubseteq t)\phi \) and *bounded universal quantifier* \( (\forall x \sqsubseteq t)\phi \) as shorthand notations for respectively

\[
\exists x \left[ x \sqsubseteq t \land \phi \right] \quad \text{and} \quad \forall x \left[ x \sqsubseteq t \rightarrow \phi \right].
\]

Next we define the \( \Sigma \)-formulas inductively by

- \( \phi \) and \( \neg \phi \) are \( \Sigma \)-formulas if \( \phi \) is of the form \( s \sqsubseteq t \), or of the form \( s = t \), where \( s \) and \( t \) are terms
- \( (\phi \lor \psi) \) and \( (\phi \land \psi) \) are \( \Sigma \)-formulas if \( \phi \) and \( \psi \) are \( \Sigma \)-formulas
- \( (\exists x \sqsubseteq t)\phi \) and \( (\forall x \sqsubseteq t)\phi \) and \( (\exists x)\phi \) are \( \Sigma \)-formulas if \( \phi \) is a \( \Sigma \)-formula, \( t \) is a term and \( x \) is a variable not occurring in \( t \).

A *purely existential formula* is a \( \Sigma \)-formula that does not contain bounded universal quantifiers.

We assume that the reader notes the similarities with first-order number theory. The formulas that correspond to \( \Sigma \)-formulas in number theory are often called \( \Sigma_1 \)-formulas or \( \Sigma_0^1 \)-formulas. Now, in contrast to in number theory, it is not clear how the relation symbol that defines the bounded quantifiers should
be interpreted. In the standard model for number theory the relation symbol should obviously be interpreted as the standard ordering relation of the natural numbers. There does not seem to be any other natural options. We can chose between the less-than or less-than-or-equal-to relation over the natural numbers, and that is it. In bit theory a number of essentially different interpretations will make sense. We might interpret $x \sqsubseteq y$ as

- $x$ is a substring of $y$
- $x$ is a prefix of $y$
- $x$ is shorter than $y$.

The three interpretations listed above are the ones that will be investigated in the current paper, but there are definitely other interesting options. We might e.g. interpret $\sqsubseteq$ as the subword relation investigated in Halfon et. al. \[8\], or as a lexicographical ordering relation.

### 1.5 More on Notation and Terminology

We reserve the letter $\mathfrak{B}$ for the structure where $\sqsubseteq$ is the substring relation, that is, the $\mathcal{L}_{BT}$-structure $\mathfrak{B}$ is the extension of $\mathfrak{B}^-$ where $\alpha \sqsubseteq^\mathfrak{B} \beta$ holds iff there exists $\gamma, \delta \in \{0, 1\}^*$ such that $\gamma \alpha \delta = \beta$.

The $\mathcal{L}_{BT}$-structure $\mathfrak{D}$ is the same structure as $\mathfrak{B}$ with one exception: the relation $\alpha \sqsubseteq^\mathfrak{D} \beta$ holds iff $\alpha$ is a prefix of $\beta$, that is, iff there exists $\gamma \in \{0, 1\}^*$ such that $\alpha \gamma = \beta$.

The $\mathcal{L}_{BT}$-structure $\mathfrak{F}$ is the same structure as $\mathfrak{B}$ with one exception: the relation $\alpha \sqsubseteq^\mathfrak{F} \beta$ holds iff the number of bits in $\alpha$ is less than or equal to the number of bits in $\beta$, that is, iff $|\alpha| \leq |\beta|$.

To improve the readability we will use the symbol $\preceq$ in place of the symbol $\sqsubseteq$ in formulas that are meant to be interpreted in the structure $\mathfrak{D}$. Thus, $x \preceq y$ should be read as “$x$ is a prefix of $y$”. Similarly, we will use $\preceq$ in formulas that are meant to be interpreted in $\mathfrak{F}$, and $x \preceq y$ should be read as “$x$ is shorter than $y$”. We will continue to use the symbol $\sqsubseteq$ in formulas that are meant to be interpreted in $\mathfrak{B}$. Thus, $x \sqsubseteq y$ should be read as “$x$ is a substring of $y$”.

We may skip the operator $\circ$ in first-order formulas and simply write $st$ in place of $s \circ t$. Furthermore, we will occasionally contract quantifiers and write, e.g., $\forall xy \sqsubseteq z[\phi]$ in place of $(\forall x \sqsubseteq z)(\forall y \sqsubseteq z)\phi$, and for $\sim \in \{\preceq, \sqsubseteq, \preceq, \preceq, =\}$, we will normally write $s \not\sim t$ in place of $\neg s \sim t$.

Recall that a sentence is a formula with no free variables.

### 1.6 References and Related Research

Formal concatenation theory can be traced back to Tarski \[23\] and Hermes \[9\] (see \[20\] for an English review). Quine \[19\] and Corcoran et al. \[2\] are also papers
on the subject that may have some historical interest. A rather recent line of research in concatenation theory has focused on interpretability between weak first-order theories and essential undecidability. Grzegorczyk [5], Grzegorczyk & Zdanowski [6], Svejdar [22], Visser [24], Horihata [11] and Higuchi & Horihata [10] belong to this line. Another recent line of research has focused on word equations and formal languages. Papers in this line, like e.g., Karhumäki et al. [12], Ganesh et al. [4], Halfon et al. [8] and Day et al. [3], are in general oriented towards theoretical computer science. Both lines are related to our research: the former to the first-order theories we present in Section 2, the latter to the results we present in last two sections of the paper. More on the history of concatenation theory can be found in Visser [24].

The present paper is a significantly extended and improved version of the conference paper Kristiansen & Murwanashyaka [13]. We assume that the reader is familiar with the basics of mathematical logic and computability theory. An introduction can be found Leary & Kristiansen [15] (Chapter 8 contains some material on concatenation theory). Some familiarity with first-order arithmetic will probably also be beneficial to the reader. An introduction can be found in Hajek & Pudlak [7].

2 Σ-Complete Axiomatizations

Once the bounded quantifiers are present, it is natural to ask if we can find neat and natural Σ-complete axiomatisations of bit theory. Corresponding axiomatisations of number theory, e.g. Robinson Arithmetic, have been of great importance in logic. In this section we will give finite sets of axioms that are Σ-complete with respect to our different structures (B, D and F). These axiomatisations might serve as base theories which can be extended with, e.g., collection principles or induction schemes of the form

\[ \left( \phi(e) \land \forall x[ \phi(x) \rightarrow (\phi(x0) \land \phi(x1)) ] \right) \rightarrow \forall x[\phi(x)] . \]

2.1 The Theory B⁻

B⁻ is a \( \mathcal{L}_{\bf B}^- \)-theory. All the first-order theories we will present contain the axioms of B⁻.

Definition 3. The first-order theory B⁻ contains the following four non-logical axioms:

1. \( \forall x[ x = ex \land x = xe ] \)
2. \( \forall xyz[ (xy)z = x(yz) ] \)
3. \( \forall xy[ (x \neq y) \rightarrow (x0 \neq y0) \land (x1 \neq y1) ] \)
4. \( \forall xy[ x0 \neq y1 ] \)
We will use $B^{-}_i$ to refer to the $i^{th}$ axiom of $B^-$. 

**Lemma 4.** Let $\alpha, \beta \in \{0, 1\}^*$. Then $B^- \vdash \overline{\alpha} \circ \overline{\beta} = \overline{\alpha \beta}$.

**Proof.** We prove the lemma by induction on the length of $\beta$, and we consider the following cases: $\beta \equiv \varepsilon$, $\beta \equiv \gamma 0$ and $\beta \equiv \gamma 1$.

Assume $\beta \equiv \varepsilon$. Then $\overline{\beta} \equiv e$, and the lemma holds by the axiom $B^{-}_1$. Assume $\beta \equiv \gamma 0$. By the induction hypothesis, we have $B^- \vdash \overline{\alpha} \circ \overline{\gamma} = \overline{\alpha \gamma}$. Thus, we also have

$$B^- \vdash (\overline{\alpha} \circ \overline{\gamma}) \circ 0 = \overline{\alpha \gamma} \circ 0 .$$

By $B^{-}_2$, we have

$$B^- \vdash \overline{\alpha} \circ (\overline{\gamma} \circ 0) = \overline{\alpha \gamma} \circ 0 .$$

Thus, the lemma holds as $\overline{\beta} \equiv \overline{\gamma} \circ 0$ and $\overline{\alpha \beta} \equiv \overline{\alpha \gamma} \equiv \overline{\alpha \gamma} \circ 0$. The case $\beta \equiv \gamma 1$ is similar.

**Lemma 5.** For any $\mathcal{L}_{BT}$-term $t$ there is $\alpha \in \{0, 1\}^*$ such that $B^- \vdash t = \overline{\alpha}$.

**Proof.** We will use induction on the structure of $t$. Assume $t \equiv e$. Obviously, $B^- \vdash e = e$. Thus, we have $B^- \vdash e = e$ as $e \equiv e$. Assume $t \equiv 0$. By $B^{-}_1$, we have $B^- \vdash 0 = e \circ 0$. Thus, we have $B^- \vdash 0 = \overline{0} = e \circ 0$. The case $t \equiv 1$ is similar to the case $t \equiv 0$. Finally, assume $t \equiv t_1 \circ t_2$. By induction hypothesis we have $\alpha_1, \alpha_2 \in \{0, 1\}^*$ such that $B^- \vdash t_1 = \overline{\alpha_1}$ and $B^- \vdash t_2 = \overline{\alpha_2}$. By Lemma 4, we have $B^- \vdash \overline{\alpha_1} \circ \overline{\alpha_2} = \overline{\alpha_1 \alpha_2}$. Thus, $B^- \vdash t = \overline{\alpha_1 \alpha_2}$. 

**Lemma 6.** $B^{-}_1, B^{-}_2, B^{-}_4 \vdash \forall x [x 0 \neq e \wedge x 1 \neq e ]$.

**Proof.** We reason in an arbitrary model for $\{B^{-}_1, B^{-}_2, B^{-}_4\}$. Let $x$ be an arbitrary element in the universe. Assume $x 0 = e$. Then $1(x 0) = 1 e$. By $B^{-}_1$, we have $1(x 0) = 1$. By $B^{-}_2$, we have $(1 x) 0 = 1$. By $B^{-}_4$, we have $(1 x) 0 = e 1$. This contradicts $B^{-}_4$. This proves that $x 0 \neq e$. A symmetric argument shows that $x 1 \neq e$.

**Lemma 7.** Let $\alpha, \beta \in \{0, 1\}^*$ and $\alpha \neq \beta$. Then $B^- \vdash \overline{\alpha} \neq \overline{\beta}$.

**Proof.** We prove the lemma by induction on the natural number $\min(|\alpha|, |\beta|)$. Assume $\min(|\alpha|, |\beta|) = 0$. Then, either $\alpha$ or $\beta$ will be the empty string $\varepsilon$. Furthermore, it cannot be the case that both $\alpha$ or $\beta$ are the empty string as $\alpha \neq \beta$. By Lemma 5, we have $B^- \vdash \overline{\alpha} \neq \overline{\beta}$.

Assume $\min(|\alpha|, |\beta|) > 0$. Then we have $\alpha \equiv \alpha'a$ and $\beta \equiv \beta'b$ where $\alpha, \beta \in \{0, 1\}^* \land a, b \in \{0, 1\}$. The proof splits into two cases: (i) $a \neq b$ and (ii) $a$ and $b$ are different. Case (i): The induction hypothesis yields $B^- \vdash \overline{\alpha'} \neq \overline{\beta'}$, and the lemma follows by $B^{-}_4$. Case (ii): The lemma follows straightaway by $B^{-}_4$ (we do not need the induction hypothesis).
We leave the proof of the next theorem to the reader (see the proof of Theorem [13]).

**Theorem 8.** For any purely existential sentence \( \varphi \), we have

\[
\mathfrak{B}^- \models \varphi \Rightarrow B^- \vDash \varphi.
\]

### 2.2 The Structure \( \mathfrak{B} \)

**Definition 9.** The first-order theory \( B \) contains the following eleven non-logical axioms:

1. the first four axioms are the axioms of \( B^- \)
2. \( \forall x \left[ x \subseteq e \leftrightarrow x = e \right] \)
3. \( \forall x \left[ x \subseteq 0 \leftrightarrow (x = e \lor x = 0) \right] \)
4. \( \forall x \left[ x \subseteq 1 \leftrightarrow (x = e \lor x = 1) \right] \)
5. \( \forall xy \left[ x \subseteq 0y0 \leftrightarrow (x = 0y0 \lor x \subseteq 0y \lor x \subseteq y0) \right] \)
6. \( \forall xy \left[ x \subseteq 0y1 \leftrightarrow (x = 0y1 \lor x \subseteq 0y \lor x \subseteq y1) \right] \)
7. \( \forall xy \left[ x \subseteq 1y0 \leftrightarrow (x = 1y0 \lor x \subseteq 1y \lor x \subseteq y0) \right] \)
8. \( \forall xy \left[ x \subseteq 1y1 \leftrightarrow (x = 1y1 \lor x \subseteq 1y \lor x \subseteq y1) \right] \)

We will use \( B_i \) to refer to the \( i \)th axiom of \( B \).

**Lemma 10.** Let \( \alpha, \beta \in \{0, 1\}^* \) and \( \alpha \sqsubseteq^B \beta \) (i.e. \( \alpha \) is a substring of \( \beta \)). Then \( B \vDash \overline{\overline{\alpha}} \subseteq \overline{\overline{\beta}} \).

**Proof.** We prove this lemma by induction on the length of \( \beta \). The proof splits into the cases \( \beta \equiv \varepsilon \), \( \beta \equiv 0 \), \( \beta \equiv 1 \), \( \beta \equiv 0 \gamma 0 \), \( \beta \equiv 0 \gamma 1 \), \( \beta \equiv 1 \gamma 0 \) and \( \beta \equiv 1 \gamma 1 \).

In the cases when \( \beta \) is empty or just contains a single bit, the lemma holds by \( B_5 \), \( B_6 \) or \( B_7 \).

Assume \( \beta \equiv 0 \gamma 0 \). The proof splits into the cases

1. \( \alpha = 0 \gamma 0 \),
2. \( \alpha \sqsubseteq^B 0 \gamma \) and
3. \( \alpha \sqsubseteq^B \gamma 0 \).

In case (i), we have \( B \vDash \overline{\alpha} = \overline{0 \gamma 0} \) by logical axioms. By \( B_8 \), we have \( B \vDash \overline{\alpha} \subseteq \overline{\gamma 0} \). In case (ii), the induction hypothesis yields \( B \vDash \overline{\alpha} \subseteq \overline{0 \gamma} \). By \( B_8 \), we have \( B \vDash \overline{\alpha} \subseteq \overline{\gamma 0} \). In case (iii), the induction hypothesis yields \( B \vDash \overline{\alpha} \subseteq \overline{\gamma 0} \). By \( B_8 \), we have \( B \vDash \overline{\alpha} \subseteq \overline{\gamma 0} \).

The cases \( \beta \equiv 0 \gamma 1 \), \( \beta \equiv 1 \gamma 0 \) and \( \beta \equiv 1 \gamma 1 \) are similar to the case \( \beta \equiv 0 \gamma 0 \), use \( B_9 \), \( B_{10} \) and \( B_{11} \), respectively, in place of \( B_8 \). □

**Lemma 11.** Let \( \alpha, \beta \in \{0, 1\}^* \) and \( \alpha \not\sqsubseteq^B \beta \) (i.e. \( \alpha \) is a not substring of \( \beta \)). Then \( B \vDash \overline{\overline{\alpha}} \not\subseteq \overline{\overline{\beta}} \).

**Proof.** The proof of this lemma is symmetric to the proof of Lemma [10] and we omit the details. □
Lemma 12. Let \( \phi(x) \) be an \( \mathcal{L}_{BT} \)-formula such that
\[
\mathfrak{B} \models \phi(\alpha) \implies B \vdash \phi(\alpha)
\]
for any \( \alpha \in \{0, 1\}^* \). Then we have
\[
\mathfrak{B} \models (\forall x \subseteq \alpha)\phi(x) \implies B \vdash (\forall x \subseteq \alpha)\phi(x)
\]
for any \( \alpha \in \{0, 1\}^* \).

Proof. We proceed by induction on the length of \( \alpha \). We will consider the cases \( \alpha \equiv \varepsilon \), \( \alpha \equiv 0 \), \( \alpha \equiv 1 \), \( \alpha \equiv 0\beta 0 \), \( \alpha \equiv 0\beta 1 \), \( \alpha \equiv 1\beta 0 \) and \( \alpha \equiv 1\beta 1 \).

Let \( \alpha \equiv \varepsilon \). Assume \( \mathfrak{B} \models (\forall x \subseteq e)\phi(x) \). Then \( \mathfrak{B} \models \phi(e) \). By (*), we have \( B \vdash \phi(e) \). By \( B_6 \), we have \( B \vdash (\forall x \subseteq e)\phi(x) \).

Let \( \alpha \equiv 0 \). Assume \( \mathfrak{B} \models (\forall x \subseteq \emptyset)\phi(x) \). Then \( \mathfrak{B} \models \phi(\emptyset) \). By (*), we have \( B \vdash \phi(\emptyset) \). By \( B_6 \), we have \( B \vdash (\forall x \subseteq \emptyset)\phi(x) \). The case \( \alpha \equiv 1 \) is similar to the case \( \alpha \equiv 0 \). Use \( B_7 \) in place of \( B_6 \).

Let \( \alpha \equiv 0\beta 0 \). Assume \( \mathfrak{B} \models (\forall x \subseteq 0\beta 0)\phi(x) \). Then
\[
\mathfrak{B} \models \phi(0\beta 0) \land (\forall x \subseteq 0\beta)\phi(x) \land (\forall x \subseteq \beta 0)\phi(x) .
\]

By (*) and the induction hypothesis, we have
\[
B \vdash \phi(0\beta 0) \land (\forall x \subseteq 0\beta)\phi(x) \land (\forall x \subseteq \beta 0)\phi(x) .
\]

By \( B_8 \), we have \( B \vdash (\forall x \subseteq 0\beta 0)\phi(x) \).

The case \( \alpha \equiv 0\beta 1 \), the case \( \alpha \equiv 1\beta 0 \) and the case \( \alpha \equiv 1\beta 1 \) are handled similarly using \( B_9 \), \( B_{10} \) and \( B_{11} \), respectively, in place of \( B_8 \).

\[\square\]

Theorem 13 (\( \Sigma \)-completeness of \( B \)). For any \( \Sigma \)-sentence \( \phi \), we have
\[
\mathfrak{B} \models \phi \iff B \vdash \phi .
\]

Proof. We prove the theorem by induction on the structure of the \( \Sigma \)-sentence \( \phi \). The base cases are \( \phi \equiv s = t \), \( \phi \equiv s \neq t \), \( \phi \equiv s \subseteq t \) and \( \phi \equiv s \nsubseteq t \) (where \( s \) and \( t \) are variable free). We attend to the case \( \phi \equiv s \subseteq t \). So assume \( \phi \equiv s \subseteq t \).

Furthermore, assume \( \mathfrak{B} \models s \subseteq t \). By Lemma 13, we have \( \alpha, \beta \in \{0, 1\}^* \) such that
\[
B \vdash s = \alpha \land t = \beta .
\]

By the Soundness Theorem for first-order logic, we have \( \alpha \subseteq \mathfrak{B} \beta \). By Lemma 10, we have \( B \vdash \alpha \subseteq \beta \). By (*), we have \( B \vdash s \subseteq t \). This proves that the theorem holds when \( \phi \equiv s \subseteq t \). The cases \( \phi \equiv s = t \), \( \phi \equiv s \neq t \) and \( \phi \equiv s \nsubseteq t \) are similar.

Use Lemma 7 in place of Lemma 10 when \( \phi \equiv s \neq t \). Use Lemma 11 in place of Lemma 10 when \( \phi \equiv s \nsubseteq t \).

We turn to the inductive cases. Let \( \phi \equiv (\psi \land \xi) \). Assume \( \mathfrak{B} \models \psi \land \xi \). Then we have \( \mathfrak{B} \models \psi \) and \( \mathfrak{B} \models \xi \). By the induction hypothesis, we have \( B \vdash \psi \) and \( B \vdash \xi \).

Thus, \( B \vdash \psi \land \xi \). The case \( \phi \equiv (\psi \lor \xi) \) is similar.
Let $\phi \equiv (\exists x)\psi(x)$. Assume $\mathfrak{B} \models (\exists x)\psi(x)$. Then we have $\mathfrak{B} \models \psi(\alpha)$ for some $\alpha \in \{0, 1\}^\ast$. Our induction hypothesis yields $B \vdash \psi(\alpha)$, and then we also have $B \vdash (\exists x)\psi(x)$.

Let $\phi \equiv (\forall x \subseteq t)\psi(x)$. Assume $\mathfrak{B} \models (\forall x \subseteq t)\psi(x)$. By Lemma 5, we have $\beta \in \{0, 1\}^\ast$ such that

$$B \vdash t = \overrightarrow{\beta} \quad (\dagger)$$

By the Soundness Theorem of first-order logic, we have

$$\mathfrak{B} \models (\forall x \subseteq \overrightarrow{\beta})\psi(x) \quad (\ddagger)$$

Our induction hypothesis yields

$$\mathfrak{B} \models \psi(\alpha) \Rightarrow B \vdash \psi(\alpha)$$

for all $\alpha \in \{0, 1\}^\ast$. By (IH), (\dagger) and Lemma 12, we have $B \vdash (\forall x \subseteq \overrightarrow{\beta})\psi(x)$. Finally, by (\ddagger), we have $B \vdash (\forall x \subseteq t)\psi(x)$. This completes the case where $\phi$ is of the form $(\forall x \subseteq t)\psi(x)$. We leave the case $\phi \equiv (\exists x \subseteq t)\psi(x)$ to the reader. \qed

### 2.3 The Structure $\mathfrak{D}$

**Definition 14.** The first-order theory $D$ contains the following seven non-logical axioms:

- the first four axioms are the axioms of $B^-$
  5. $\forall x[ x \preceq e \iff x = e ]$
  6. $\forall xy[ x \preceq y0 \iff (x = y0 \lor x \preceq y) ]$
  7. $\forall xy[ x \preceq y1 \iff (x = y1 \lor x \preceq y) ]$

We will use $D_i$ to refer to the $i^{th}$ axiom of $D$.

**Lemma 15.** Let $\alpha, \beta \in \{0, 1\}^\ast$ and $\alpha \preceq^D \beta$ (i.e. $\alpha$ is a prefix of $\beta$). Then $D \vdash \alpha \preceq \overrightarrow{\beta}$.

**Proof.** This proof is symmetric to the proof of the next lemma. We omit the details. \qed

**Lemma 16.** Let $\alpha, \beta \in \{0, 1\}^\ast$ and $\alpha \not\preceq^D \beta$ (i.e. $\alpha$ is a not prefix of $\beta$). Then $D \vdash \alpha \not\preceq \overrightarrow{\beta}$.

**Proof.** We prove the lemma by induction on the length of $\beta$, and we consider the cases $\beta \equiv \varepsilon$, $\beta \equiv \gamma 0$ and $\beta \equiv \gamma 1$.

Assume $\beta$ is empty. Then $\alpha$ is not empty. By Lemma 7, we have $D \vdash \alpha \not\preceq e$. By axiom $D_5$, we have $D \vdash \alpha \not\preceq e$. Thus, $D \vdash \alpha \not\preceq \overrightarrow{\beta}$. Assume $\beta \equiv \gamma 0$. Then we have $\alpha \not\preceq \gamma 0$ and $\alpha \not\preceq \gamma \overrightarrow{\beta}$. By our induction hypothesis, we have $D \vdash \alpha \not\preceq \overrightarrow{\gamma}$. By Lemma 7, we have $D \vdash \alpha \not\preceq \overrightarrow{\gamma 0}$. By axiom $D_6$, we have $D \vdash \alpha \not\preceq \overrightarrow{\gamma 0}$. The case where $\beta$ is of the form $\gamma 1$ is similar. Apply axiom $D_7$ in place of $D_6$. \qed
Lemma 17. Let $\phi(x)$ be an $\mathcal{L}_{BT}$-formula such that

$$\mathcal{D} \models \phi(\overline{a}) \Rightarrow D \vdash \phi(\overline{a})$$

(*)

for any $\alpha \in \{0, 1\}^\ast$. Then we have

$$\mathcal{D} \models (\forall x \leq \overline{a})\phi(x) \Rightarrow D \vdash (\forall x \leq \overline{a})\phi(x)$$

for any $\alpha \in \{0, 1\}^\ast$.

Proof. We prove the lemma by induction on the length of $\alpha$, and we consider the cases $\alpha \equiv \varepsilon$, $\alpha \equiv \beta_0$ and $\alpha \equiv \beta_1$.

Let $\alpha \equiv \varepsilon$. Assume $\mathcal{D} \models (\forall x \leq e)\phi(x)$. Then we have $\mathcal{D} \models \phi(e)$. By (*), we have $D \vdash \phi(e)$. By $D_5$, we have $D \vdash (\forall x \leq e)\phi(x)$.

Let $\alpha \equiv \beta_0$. Assume $\mathcal{D} \models (\forall x \leq \beta_0)\phi(x)$. Then we have $\mathcal{D} \models (\forall x \leq \beta)\phi(x)$ and $\mathcal{D} \models \phi(\overline{\beta_0})$. By our induction hypothesis we have $D \vdash (\forall x \leq \beta)\phi(x)$. By (*), we have $D \vdash \phi(\overline{\beta_0})$. By axiom $D_6$ we have $D \vdash (\forall x \leq \beta_0)\phi(x)$.

The case $\alpha \equiv \beta_1$ is similar to the preceding case. Use $D_7$ in place of $D_6$. \qed

Theorem 18 (Σ-completeness of $D$). For any $\Sigma$-sentence $\phi$, we have

$$\mathcal{D} \models \phi \Rightarrow D \vdash \phi$$

Proof. Given the lemmas above, we can more or less just repeat the proof of Theorem 13. Use Lemma 15 in place of Lemma 10, Lemma 16 in place of Lemma 11 and Lemma 17 in place of Lemma 12. \qed

2.4 The Structure $\mathcal{F}$

Definition 19. The first-order theory $F$ contains the following eleven non-logical axioms:

- the first four axioms are the axioms of $B^-$

5. $\forall x[ e \leq x ]$
6. $\forall x[ x \leq e \rightarrow x = e ]$
7. $\forall xy[ x0 \leq y0 \leftrightarrow x \leq y ]$
8. $\forall xy[ x0 \leq y1 \leftrightarrow x \leq y ]$
9. $\forall xy[ x1 \leq y0 \leftrightarrow x \leq y ]$
10. $\forall xy[ x1 \leq y1 \leftrightarrow x \leq y ]$
11. $\forall x[ x = e \lor \exists y[ x = y0 \lor x = y1 ] ]$.

We will use $F_i$ to refer to the $i^{th}$ axiom of $F$.

Lemma 20. Let $\alpha, \beta \in \{0, 1\}^\ast$ and $\alpha \preceq \beta$ (i.e. $|\alpha| \leq |\beta|$). Then $F \vdash \overline{\alpha} \trianglelefteq \overline{\beta}$. 
Lemma 21. Let \( \alpha, \beta \in \{0, 1\}^* \) and \( \alpha \not\preceq^\beta \beta \) (i.e. \( |\alpha| > |\beta| \)). Then \( F \vdash \sigma \not\preceq \beta \).

Proof. We prove this lemma by induction on the length of \( \beta \), and we consider the cases \( \alpha \equiv \varepsilon \), \( \alpha \equiv \alpha'0 \) and \( \alpha \equiv \alpha'1 \).

If \( \alpha \equiv \varepsilon \), we have \( F \vdash \sigma \preceq \beta \) by \( F_5 \). Let \( \alpha \equiv \alpha'0 \). Since \( |\alpha| \leq |\beta| \), we have \( b \in \{0, 1\} \) and \( \beta' \in \{0, 1\}^* \) such that \( \beta \equiv \beta'b \). The induction hypothesis yields \( F \vdash \alpha' \preceq \beta' \). If \( b = 0 \), we have \( F \vdash \sigma \preceq \beta \) by \( F_7 \). If \( b = 1 \), we have \( F \vdash \sigma \preceq \beta \) by \( F_8 \). This proves the case \( \alpha \equiv \alpha'0 \). The proof when \( \alpha \equiv \alpha'1 \) is similar. Use \( F_0 \) and \( F_{10} \), respectively, in place of \( F_7 \) and \( F_8 \).

It is convenient to introduce some new notation before we state our next lemma: For any \( \alpha \in \{0, 1\}^* \), let

\[
[\varepsilon \ldots \alpha] = \{ \beta \mid \beta \in \{0, 1\}^* \text{ and } \beta \preceq^\beta \alpha \}.
\]

Lemma 22. We have

\[
F \vdash \forall x[ x \preceq \sigma \rightarrow \bigvee_{\beta \in [\varepsilon \ldots \alpha]} x = \beta ]
\]

for any \( \alpha \in \{0, 1\}^* \).

Proof. We prove this lemma by induction on the length of \( \alpha \). The base case is \( \alpha \equiv \varepsilon \). The inductive cases are \( \alpha \equiv \gamma 0 \) and \( \alpha \equiv \gamma 1 \).

First we deal with case \( \alpha \equiv \varepsilon \). The axiom \( F_0 \) says that

\[
\forall x[ x \preceq e \rightarrow x = e ].
\]

Thus, we have

\[
F \vdash \forall x[ x \preceq \sigma \rightarrow \bigvee_{\beta \in [\varepsilon \ldots \alpha]} x = \beta ]
\]

straightaway as \( \sigma \) is \( e \), the set \([\varepsilon \ldots \alpha]\) is the singleton set \( \{\varepsilon\} \) and \( \pi \) is \( e \).
We will now turn to the inductive case $\alpha \equiv \gamma 0$. In this case it is sufficient to prove

\[ F \vdash \forall x [ x = e \rightarrow ( x \leq \gamma 0 \rightarrow \bigvee_{\beta \in [e \ldots \gamma 0]} x = \beta ) ] \]

\[ F \vdash \forall xy [ x = y 0 \rightarrow ( x \leq \gamma 0 \rightarrow \bigvee_{\beta \in [e \ldots \gamma 0]} x = \beta ) ] \]

\[ F \vdash \forall xy [ x = y 1 \rightarrow ( x \leq \gamma 0 \rightarrow \bigvee_{\beta \in [e \ldots \gamma 0]} x = \beta ) ] . \]

as it follows from (1), (2), (3) and the axiom $F_{11}$ that

\[ F \vdash \forall x [ x \leq \gamma 0 \rightarrow \bigvee_{\beta \in [e \ldots \gamma 0]} x = \beta ] . \]

Our induction hypothesis yields

\[ F \vdash \forall x [ x \leq \gamma \rightarrow \bigvee_{\beta \in [e \ldots \gamma]} x = \beta ] . \]  \hspace{1cm} \text{(IH)}

It is trivial that (1) holds. This is a logical truth that holds in any model. We do not need any of our non-logical axioms to prove (1). Let us turn to the proof of (2). We reason in an arbitrary model for $F$. Assume $x = y 0$ and $x \leq \gamma 0$. We need to argue that

\[ \bigvee_{\beta \in [e \ldots \gamma 0]} x = \beta \]  \hspace{1cm} \text{(*)}

holds in the model. It is obvious that we have $y 0 \leq \gamma 0$. By $F_7$, we have $y \leq \gamma$. By (IH), we have

\[ \bigvee_{\beta \in [e \ldots \gamma]} y = \beta . \]

Thus, we also have

\[ \bigvee_{\beta \in [e \ldots \gamma]} y 0 = \beta 0 \]

as $y = \beta \rightarrow y 0 = \beta 0$. Furthermore, since $x = y 0$ and $\beta 0 \equiv \beta 0$, we have

\[ \bigvee_{\beta \in [e \ldots \gamma]} x = \beta 0 . \]  \hspace{1cm} \text{(**)}

Finally, we observe that (**)) implies (*). This proves (2).

The proof of (3) is symmetric to the proof of (2). Use the axiom $F_9$ in place of $F_7$. This completes the proof for the case $\alpha \equiv \gamma 0$. The case $\alpha \equiv \gamma 1$ is symmetric.

Use the axioms $F_8$ and $F_{10}$, respectively, in place of $F_7$ and $F_9$. \qed
Lemma 23. Let $\phi(x)$ be an $L_{BT}$-formula such that

$$\mathfrak{F} \models \phi(\overline{\alpha}) \Rightarrow F \vdash \phi(\overline{\alpha})$$

for any $\alpha \in \{0,1\}^\ast$. Then we have

$$\mathfrak{F} \models (\forall x \leq \overline{\alpha}) \phi(x) \Rightarrow F \vdash (\forall x \leq \overline{\alpha}) \phi(x)$$

for any $\alpha \in \{0,1\}^\ast$.

Proof. Assume $\mathfrak{F} \models (\forall x \leq \overline{\alpha}) \phi(x)$. Then, we have

$$\mathfrak{F} \models \bigwedge_{\beta \in [\varepsilon \ldots \alpha]} \phi(\overline{\beta}) .$$

By (*), we have

$$F \vdash \bigwedge_{\beta \in [\varepsilon \ldots \alpha]} \phi(\overline{\beta}) .$$

By Lemma 22, we have $F \vdash (\forall x \leq \overline{\alpha}) \phi(x)$.

Theorem 24 (\Sigma- completeness of $F$). For any $\Sigma$-sentence $\phi$, we have

$$\mathfrak{F} \models \phi \Rightarrow F \vdash \phi .$$

Proof. Given the lemmas above, we can more or less just repeat the proof of Theorem 13 Use Lemma 23 in place of Lemma 10, Lemma 21 in place of Lemma 11 and Lemma 23 in place of Lemma 12.

Both $B$ and $D$ are open theories (all the axioms are purely universal statements) whereas $F$ is not. The axiom $F_{11}$ contains an existential quantifier. Can we find a purely universal set of axioms that is $\Sigma$-complete with respect to the model $\mathfrak{F}$? Yes, we can. We can regard Lemma 22 as an axiom scheme. Then we do not need $F_{11}$ to achieve $\Sigma$-completeness.

Definition 25. Let $F'$ be the first-order theory $F$ where the axiom $F_{11}$ is replaced by the scheme

$$\forall x[ x \leq \overline{\alpha} \rightarrow \bigvee_{\beta \in [\varepsilon \ldots \alpha]} x = \overline{\beta} ] .$$

(for $\alpha \in \{0,1\}^\ast$)

Theorem 26 (\Sigma- completeness of $F'$). For any $\Sigma$-sentence $\phi$, we have

$$\mathfrak{F} \models \phi \Rightarrow F' \vdash \phi .$$

Proof. Proceed as in the proof of Theorem 24. Since the axiom scheme is present, $F_{11}$ will not be needed anymore.

Now, $F'$ is an open theory, but in contrast to $B$ and $D$, it is not finite.
Conjecture: There is no finite open set of axioms that is $\Sigma$-complete with respect to the structure $\mathfrak{F}$.

3 Normal Form Theorems

After we have endowed bit theory with bounded quantifiers, it becomes natural to search for normal forms and see if we can prove normal form theorems similar to the ones we know from number theory.

Some lemmas (27, 29, 30) in this section can be found elsewhere, e.g., in Büchi & Senger [1], Senger [21] and Karhumäki et al. [12]. We have given complete proofs below in order to make our paper self-contained (the proofs we give do not differ essentially from those given in Büchi & Senger [1]).

The next lemma shows that conjunctions of equations can be replaced by one equation.

Lemma 27. Let $s_1, s_2, t_1, t_2$ be $L_{BT}$-terms. We have

$$\mathfrak{B}^- \models (s_1 = t_1 \land s_2 = t_2) \leftrightarrow s_1 s_2 s_1 s_2 = t_1 t_2 t_1 t_2.$$  

Proof. Let $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \{0, 1\}^*$. Assume $\alpha_1 0 \alpha_2 1 \alpha_2 = \beta_1 0 \beta_1 0 \beta_1 1 \beta_2$. Then $|\alpha_1 0 \alpha_2| = |\beta_1 0 \beta_2|$ and $|\alpha_1 1 \alpha_2| = |\beta_1 1 \alpha_2|$. The proof splits into the two cases $|\alpha_1| = |\beta_1|$ and $|\alpha_2| = |\beta_2|$. Assume $|\alpha_1| \neq |\beta_1|$. We can w.l.o.g. assume that $|\alpha_1| < |\beta_1|$. This implies that

$$0 = (\alpha_1 0 \alpha_2)_{|\alpha_1|+1} = (t)_{|\alpha_1|+1} = (\alpha_1 1 \alpha_2)_{|\alpha_1|+1} = 1.$$  

This is a contradiction. This proves the implication from the right to the left. The converse implication is obvious. \hfill \square

The next lemma shows that disjunctions of equations can be replaced by one equation at the price of some more existential quantifiers.

Lemma 28. Let $s_1, s_2, t_1, t_2$ be $L_{BT}$-terms. There exist $L_{BT}$-terms $s, t$ and variables $v_1, \ldots, v_k$ such that

$$\mathfrak{D} \models (s_1 \leq t_1 \lor s_2 \leq t_2) \leftrightarrow \exists v_1 \ldots v_k [s = t].$$  

Proof. Let $x_1, \ldots, x_6$ be variables that do not occur in any of the terms $s_1, s_2, t_1, t_2$. It is not very hard to see that the formula $s_1 \leq t_1 \lor s_2 \leq t_2$ is $\mathfrak{D}$-equivalent to the formula

$$\exists x_1 \ldots x_6 [s_1 = x_1 x_2 \land t_1 = x_1 x_3 \land s_2 = x_4 x_5 \land t_2 = x_4 x_6 \land (x_2 = e \lor x_5 = e)].$$  

(*)
We will show that the disjunction in (*), that is \( x_2 = e \lor x_5 = e \), can be replaced by a formula
\[
x_2x_5 = x_5x_2 \land \exists y_1 \ldots y_4[ \quad v_1 = v'_1 \land v_2 = v'_2 \land v_3 = v'_3 \quad ]
\]
where \( v_1, v_2, v_3, v'_1, v'_2, v'_3 \) are terms. Thus, by Lemma 27 which allow us to merge conjunctions of equations, (*) will be equivalent to a formula of the form
\[
\exists x_1 \ldots x_6y_1 \ldots y_4[ \quad s = t \quad ]
\]
and our proof will be complete.

Let \( \psi(u, w) \) be the formula
\[
\exists y_1y_2y_3y_4[ \quad y_1y_2 = 0 \land y_3y_4 = 1 \land uy_1wy_2 = wy_2uy_1 \\
\quad \land uy_3wy_4 = wy_4uy_3 \quad ].
\]

We claim that
\[
\mathcal{B}^- \models (u = e \lor w = e) \leftrightarrow (uw = wu \land \psi(u, w)) \quad (**)
\]

We prove (**). Assume that \( u = e \lor w = e \) (we reason in \( \mathcal{B}^- \)). Let us say that \( u = e \) (the case when \( w = e \) is symmetric). It is obvious that we have \( uw = wu \). Moreover, \( \psi(u, w) \) holds with \( y_1 = y_3 = e, y_2 = 0 \) and \( y_4 = 1 \). This prove the left-right implication of (**).

To see that the converse implication holds, assume that \( \neg(u = e \lor w = e) \), that is, both \( u \) and \( w \) are different from the empty string. Furthermore, assume that \( uw = wu \). We will argue that \( \psi(u, w) \) does not hold: Since \( uw = wu \) and both \( u \) and \( w \) contain at least one bit, it is either the case that 0 is the last bit of both strings, or it is that case that 1 is the last bit of both strings. If 0 is the last bit of both, the two equations \( uy_3wy_4 = wy_4uy_3 \) and \( y_3y_4 = 1 \) cannot be satisfied simultaneously. If 1 is the last bit of both, the two equations \( uy_1wy_2 = wy_2uy_1 \) and \( y_1y_2 = 0 \) cannot be satisfied simultaneously. Hence we conclude that \( \psi(u, w) \) does not hold. This completes the proof of (**).

As explained above, our lemma follows from (*) and (**) by Lemma 27. □

Karhumäki et al. [12] prove that the next lemma indeed holds with \( k = 2 \).

**Lemma 29.** Let \( s_1, s_2, t_1, t_2 \) be \( L_{BT} \)-terms. There exist \( L_{BT} \)-terms \( s, t \) and variables \( v_1, \ldots, v_k \) such that
\[
\mathcal{B}^- \models (s_1 = t_1 \lor s_2 = t_2) \leftrightarrow \exists v_1 \ldots v_k[s = t] \quad .
\]

**Proof.** Observe that \( s_1 = t_1 \lor s_2 = t_2 \) is \( \mathcal{O} \)-equivalent to
\[
(s_1 \preceq t_1 \land t_1 \preceq s_1) \lor (s_2 \preceq t_2 \land t_2 \preceq s_2)
\]

which again is (logically) equivalent to

\[(s_1 \preceq t_1 \lor s_2 \preceq t_2) \land (s_1 \preceq t_1 \lor t_2 \preceq s_2) \land
   (t_1 \preceq s_1 \lor s_2 \preceq t_2) \land (t_1 \preceq s_1 \lor t_2 \preceq s_2).\]

By Lemma 27 and Lemma 28 we have terms \(s, t\) and variables \(v_1 \ldots v_k\) such that

\[D \models (s_1 = t_1 \lor s_2 = t_2) \leftrightarrow \exists v_1 \ldots v_k[s = t].\]

Thus, the lemma holds as we are dealing with a \(\mathcal{L}_{BT}\)-formula. \(\square\)

**Lemma 30.** Let \(s_1, t_1\) be \(\mathcal{L}_{BT}\)-terms. There exist \(\mathcal{L}_{BT}\)-terms \(s, t\) and variables \(v_1, \ldots, v_k\) such that

\[\mathfrak{B}^- \models \lnot s_1 \not\preceq t_1 \leftrightarrow \exists v_1 \ldots v_k[s = t].\]

**Proof.** Observe that the formula \(s \neq t\) is \(\mathfrak{B}^-\)-equivalent to the formula

\[\exists xyz[s = t_0 x \lor s = t_1 x \lor t = s_0 x \lor t = s_1 x \lor
   (s = x_1 y \land t = x_0 z) \lor (s = x_0 y \land t = x_1 z)].\]

Thus, the lemma follows from Lemma 29 and Lemma 27. \(\square\)

**Lemma 31.** Let \(s_1, t_1\) be \(\mathcal{L}_{BT}\)-terms. There exist \(\mathcal{L}_{BT}\)-terms \(s, t\) and variables \(v_1, \ldots, v_k\) such that

\[D \models s_1 \not\preceq t_1 \leftrightarrow \exists v_1 \ldots v_k[s = t].\]

**Proof.** The formula \(s_1 \not\preceq t_1\) is \(D\)-equivalent to the formula

\[(\exists u[t_1 u = s_1] \land t_1 \neq s_1) \lor \exists xyz[(t_1 = x_0 y \land s_1 = x_1 z) \lor
   (t_1 = x_1 y \land s_1 = x_0 z)].\]

Thus, the lemma follows from the preceding lemmas. \(\square\)

**Theorem 32 (Normal Form Theorem for \(\mathfrak{D}\)).** Any \(\Sigma\)-formula \(\phi\) is \(\mathfrak{D}\)-equivalent to a \(\mathcal{L}_{BT}\)-formula of the form

\[\left(Q_1^{t_1} v_1 \ldots Q_m^{t_m} v_m\right) s = t\]

where \(t_1, \ldots, t_m\) are \(\mathcal{L}_{BT}\)-terms and \(Q_j^{t_j} v_j \in \{\exists v_j, \forall v_j\} \lor \{t_j, \forall v_j \subseteq t_j\}\) for \(j = 1, \ldots, m\). Moreover, if \(\phi\) is a purely existential formula, then \(Q_j^{t_j} v_j\) is \(\exists v_j\).

**Proof.** We proceed by induction on the structure of the \(\Sigma\)-formula \(\phi\) (throughout the proof we reason in the structure \(\mathfrak{D}\)). Assume \(\phi \equiv s \preceq t\). Then \(\phi\) is equivalent to a formula of the form \(\exists x[sx = t]\) and the theorem holds. Assume \(\phi \equiv s \not\preceq t\).
Then the theorem holds by Lemma 31. Assume $\phi \equiv s \neq t$. Then the theorem holds by Lemma 30. The theorem holds trivially when $\phi$ is of the form $s = t$.

Suppose $\phi$ is of the form $\psi \wedge \xi$. By our induction hypothesis, we have formulas

$$(Q_1^{x_1}x_1) \ldots (Q_k^{x_k}x_k) s_1 = t_1 \quad \text{and} \quad (Q_1^{y_1}y_1) \ldots (Q_m^{y_m}y_m) s_2 = t_2$$

which are equivalent to respectively $\psi$ and $\xi$. Thus, $\phi$ is equivalent to a formula of the form $(Q_1^{x_1}x_1) \ldots (Q_k^{x_k}x_k)(Q_1^{y_1}y_1) \ldots (Q_m^{y_m}y_m)(s_1 = t_1 \wedge s_2 = t_2)$. By Lemma 27, we have a formula of the desired form which is equivalent to $\phi$. The case when $\phi$ is of the form $\psi \vee \xi$ is similar. Use Lemma 29 in place of Lemma 27.

The theorem follows trivially from the induction hypothesis when $\phi$ is of one of the forms $(\exists v)\psi$, $(\forall v \leq t)\psi$ and $(\exists v \leq t)\psi$.

If $\phi$ is a purely existential formula, there will be no bounded universal quantifiers among $(Q_1^{x_1}v_1) \ldots (Q_m^{y_m}v_m)$. Thus, $\phi$ is equivalent to a formula of the form

$$\exists v_1 \ldots v_k[ s_1 \leq t_1 \wedge \ldots \wedge s_\ell \leq t_\ell \wedge s = t ]$$

which again is equivalent to a formula of the form

$$\exists v_1 \ldots v_k x_1 \ldots x_\ell[ s_1 x_1 = t_1 \wedge \ldots \wedge s_\ell x_\ell = t_\ell \wedge s = t ].$$

By Lemma 27 we can conclude that any purely existential formula is equivalent to a formula of the form $\exists v_1 \ldots v_m [s = t]$.

**Theorem 33 (Normal Form Theorem for $\mathfrak{S}$).** Any $\Sigma$-formula $\phi$ is $\mathfrak{S}$-equivalent to a $\mathcal{L}_{BT}$-formula of the form

$$(\exists v_0)(Q_1 v_1 \leq t_1) \ldots (Q_m v_m \leq t_m) s = t$$

where $Q_j \in \{\exists, \forall\}$ for $j = 1, \ldots, m$. Moreover, if $\phi$ is a purely existential formula, then $\phi$ is equivalent to a formula of the form

$$\exists v_0 \ldots v_m[ v_1 \leq t_1 \wedge \ldots \wedge v_m \leq t_m \wedge s = t ].$$

**Proof.** We prove the theorem by induction on the structure of the $\Sigma$-formula $\phi$ (throughout the proof we reason in the structure $\mathfrak{S}$). Assume $\phi \equiv s \leq t$. Then $\phi$ is equivalent to a formula of the form $\exists x \leq t[x = s]$, and the theorem holds. Assume $\phi \equiv s \not\leq t$. Then $\phi$ is equivalent to $t \circ 0 \leq s$, and thus also equivalent to a formula of the form $\exists x \leq s[x = t \circ 0]$. Hence the theorem holds. Furthermore, the theorem holds by Lemma 30 when $\phi \equiv s \neq t$, and the theorem holds trivially when $\phi \equiv s = t$.

The inductive cases $\phi \equiv \psi \wedge \xi$ and $\phi \equiv \psi \vee \xi$ are similar to the corresponding cases in the proof of Theorem 32 (normal form theorem for $\mathfrak{D}$).

The cases $\phi \equiv (\exists x)\psi$, $\phi \equiv (\exists x \leq t)\psi$ and $\phi \equiv (\forall x \leq t)\psi$ are easy to deal with. A formula of the form

$$(\forall x \leq t)(\exists y)\psi$$

is equivalent to a formula of the form $(\exists z)(\forall x \leq t)(\exists y \leq z)\psi$. The cases $\phi \equiv (\forall x)\psi$, $\phi \equiv (\forall x \leq t)\psi$ and $\phi \equiv (\exists x \leq t)\psi$ are similar.
– $(\exists x \leq t)(\exists y)\psi$ is equivalent to a formula of the form $(\exists y)(\exists x \leq t)\psi$
– $(\exists x)(\exists y)\psi$ is equivalent to a formula of the form $(\exists y)(\exists x \leq z)(\exists y \leq z)\psi$.

Thus, any $\Sigma$-formula is $\exists$-equivalent to a $\Sigma$-formula that contains maximum one unbounded existential quantifier.

If $\phi$ is a purely existential formula, then $Q_1, \ldots, Q_m$ will all be existential quantifiers. Thus, it is easy to see that any purely existential formula is equivalent to a formula of the form

$$\exists v_0 \ldots v_m[ v_1 \leq t_1 \land \ldots \land v_m \leq t_m \land s = t ] .$$

It is not true that any purely existential formula is $\forall$-equivalent to a formula of the form $\exists x_1 \ldots x_n[s = t]$. This follows from the results in Karhumäki et al. [12].

E.g., a formula like $x \leq y \land y \leq x$ which states that the length of $x$ equals the length of $y$, is not $\exists$-equivalent to a formula of the form $\exists x_1 \ldots x_n[s = t]$. See Example 27 in Section 6 of [12].

Lemma 34. Let $s_1, t_1$ be $L_{BT}$-terms. There exist $L_{BT}$-terms $s, t$ and variables $v_1, \ldots, v_k$ such that

$$\mathfrak{B} \models s_1 \not\leq t_1 \iff \forall v_1 \leq t_1 \exists v_2 \ldots v_k[ s = t ] .$$

Proof. Observe that $s_1 \not\leq t_1$ is $\mathfrak{B}$-equivalent to $(\forall v \leq t_1)\alpha$ where $\alpha$ is

$$\exists x[ t_1 x = vs_1 \land x \neq e ] \lor \exists xy[ (t_1 = x0y \land vs_1 = x1z) \lor (t_1 = x1y \land vs_1 = x0z) ] .$$

If we let $vs_1 \leq t_1$ abbreviate $\exists x[vs_1 x = t_1]$, then $\alpha$ can be written as $vs_1 \not\leq t_1.$

Thus, the lemma follows by Lemma [31].

Theorem 35 (Normal Form Theorem for $\mathfrak{B}$). Any $\Sigma$-formula $\phi$ is $\mathfrak{B}$-equivalent in to a $L_{BT}$-formula of the form

$$(\exists v_0)(Q_1 v_1 \leq t_1)\ldots(Q_m v_m \leq t_m) s = t$$

where $Q_j \in \{\exists, \forall\}$ for $j = 1, \ldots, m$.

Proof. Proceed by induction on the structure of the $\Sigma$-formula $\phi$. This proof is similar to the proof of Theorem [33] (normal form theorem for $\mathfrak{F}$). If $\phi$ is of form $s \leq t$, then $\phi$ is $\mathfrak{B}$-equivalent a formula of the form $\exists x_1 x_2[x_1 s x_2 = t]$. If $\phi$ is of the form $s \not\leq t$, then then Lemma [34] says that $\phi$ is $\mathfrak{B}$-equivalent to a formula of the form $\forall v_1 \leq t_1 \exists v_2 \ldots v_k[s = t]$. The remaining cases of the inductive proof are similar to their respective cases in the proof of Theorem [33].
We have not been able to find an interesting normal form for purely existential formulas which is stronger than the one for $\Sigma$-formulas in $\mathcal{B}$. It follows from the results in Karhumäki et al. [12] that the relation $x \not\preceq \mathcal{B} y$ cannot be defined in the structure $\mathcal{B}^-$. By Theorem 16 in [12], it is not possible to define the language

$$L = \{ \alpha \mid \alpha \in \{0,1\}^* \text{ and } 101 \not\preceq \mathcal{B} \alpha \}$$

by a word equation. If we could define $x \not\preceq \mathcal{B} y$ in $\mathcal{B}^-$, then it would have been possible to define $L$ by a word equation. We refer the reader to the paper itself for further details as Theorem 16 is a rather involved statement.

Since it is not possible to define $x \not\preceq \mathcal{B} y$ in $\mathcal{B}^-$, the normal form theorem for purely existential formulas in $\mathcal{D}$ (Theorem 32), will for sure be false with respect to $\mathcal{B}$. So will the normal form theorem for purely existential formulas in $\mathcal{F}$ (Theorem 33).

4 Decidability and Undecidability

4.1 Fragments

Let us start to track the border between the decidable and the undecidable. On the one hand, we have Makanin’s result (Theorem 2). We know that it is decidable if a sentence of the form $\exists \vec{x}[s = t]$ holds in the standard model. We also know that any purely existential formula is $\mathcal{D}$-equivalent to formula of this form. On the other hand it is not very hard to prove a stronger version of Theorem 1:

**Theorem 36 (Σ-Definability of Computable Functions).** For any (partially) computable function $f : \mathbb{N}^n \to \mathbb{N}$ there exists a $\Sigma$-formula $\phi(x_1, \ldots, x_n, y)$ such that

$$f(a_1, \ldots, a_n) = b \iff \mathcal{D} \models \phi(1^{a_1}, \ldots, 1^{a_n}, 1^b).$$

Moreover, given a definition of the function, we can compute the formula.

With some effort, the interested reader should be able to accomplish a proof of this theorem. The theorem implies that it is undecidable if a $\Sigma$-sentence holds in $\mathcal{D}$. It is easy to define the relations $\preceq \mathcal{D}$ and $\not\preceq \mathcal{D}$ by $\Sigma$-formulas in the structures $\mathcal{B}$ and $\mathcal{F}$, and the bounded quantifiers of $\mathcal{D}$ can be expressed by $\Sigma$-formulas in $\mathcal{B}$ and $\mathcal{F}$, e.g., if

$$\mathcal{D} \models \phi \iff \mathcal{B} \models \phi'$$

then

$$\mathcal{D} \models (\exists x \leq t) \phi \iff \mathcal{B} \models (\exists xy \subseteq t)[xy = t \land \phi']$$

Thus Theorem 36 also implies that it undecidable if a $\Sigma$-sentence holds in $\mathcal{B}$ or $\mathcal{F}$.

In order to gain further insight into what we can—and cannot—decide in bit theory, we need to keep track of the number and the type of quantifiers that
appear in our $\Sigma$-formulas. We will say that a $\Sigma$-formula is a $\Sigma_{n,m,k}$-formula if it contains $n$ unbounded existential quantifiers, $m$ bounded existential quantifiers and $k$ bounded universal quantifiers. The fragment $\Sigma_{n,m,k}$ is the set of $\Sigma_{n,m,k}$-sentences that are true in the $L_{BT}$-structure $\mathfrak{A}$. The (purely existential) fragment $\exists^\mathfrak{A}$ is the set of purely existential sentences that are true in the $L_{BT}$-structure $\mathfrak{A}$ (recall that a purely existential formula is a $\Sigma$-formula with no occurrences of bounded universal quantifiers). A $\Delta$-formula is a $\Sigma$-formula that contain no unbounded existential quantifiers, and the fragment $\Delta^\mathfrak{A}$ is the set of $\Delta$-sentences that are true in the $L_{BT}$-structure $\mathfrak{A}$. Note that

$$\exists^\mathfrak{A} = \bigcup_{n,m \in \mathbb{N}} \Sigma_{n,m,0} \quad \text{and} \quad \Delta^\mathfrak{A} = \bigcup_{m,k \in \mathbb{N}} \Sigma_{0,m,k}.$$  

4.2 Decidable Fragments

**Theorem 37.** The fragments $\Delta^D$, $\Delta^B$ and $\Delta^F$ are decidable.

**Proof.** We prove that $\Delta^B$ is decidable. Let $\phi$ be a $\Delta$-formula. The negation of a $\Delta$-formula is logically equivalent to a $\Delta$-formula (use De Morgan’s laws). We can compute a $\Delta$-formula $\phi'$ which is logically equivalent to $\neg \phi$. By Theorem 13 we have $B \models \phi$ if $B \models \neg \phi$. By the same theorem, we have $B \models \phi'$ if $B \models \neg \phi$. The set of formulas derivable from the axioms of $B$ is computably enumerable. Hence it is decidable if $\phi$ is true in $B$.

We can prove that the fragments $\Delta^D$ and $\Delta^F$ are decidable in the same way as we also have $\Sigma$-complete axiomatizations for both $D$ and $F$. \square

In some sense, we kill a fly with a hammer when we use $\Sigma$-completeness to prove the preceding theorem. It can of course be checked by brute-force algorithms if $\Delta$-sentences are true in the structures $D$, $B$ and $F$.

**Theorem 38.** The fragment $\exists^D$ is decidable.

**Proof.** Theorem 32 states that any purely existential formula $\phi$ is $D$-equivalent to a formula of the normal form $\exists v_1 \ldots v_m | s = t$. Our proofs show that there is an algorithm for transforming $\phi$ into this normal form. Thus the theorem follows by Makanin’s result (Theorem 2). \square

*Open Problem:* Is the fragment $\exists^B$ decidable?

*Open Problem:* Is the fragment $\exists^F$ decidable?

4.3 The Modulo Problem

In 13 we use the Post’s Correspondence Problem (Post 15) to prove that the fragments $\Sigma^D_{3,0,2}$, $\Sigma^D_{4,1,1}$, $\Sigma^B_{1,2,1}$ and $\Sigma^B_{1,0,2}$ are undecidable. We will improve
these results considerably in the next section. In this section we introduce an undecidable problem that is especially tailored for our needs in that respect, that is, an undecidable problem that makes the proofs in next section smooth and transparent. In lack of a better name, we dub the problem the \emph{Modulo Problem}.

We assume that the reader is familiar with modulo arithmetic and elementary number theory. We use $f^N$ to denote the $N$th iteration of an unary function $f$, that is, $f^0(X) = X$ and $f^{N+1}(x) = f(f^N(x))$.

**Definition 39.** The Modulo Problem is given by

- \textbf{Instance:} a list of pairs $\langle A_0, B_0 \rangle, \ldots, \langle A_{M-1}, B_{M-1} \rangle$ where $M > 1$ and $A_i, B_i \in \mathbb{N}$ (for $i = 0, \ldots, M-1$).
- \textbf{Solution:} a natural number $N$ such that $f^N(3) = 2$ where

$$f(x) = A_j z + B_j$$

if there exists $j < M$ such that $x = zM + j$.

The undecidability of the Modulo Problem follows from the existence of a certain type of Collatz functions constructed in Kurtz & Simon \[14\]. We will spend the rest of this section to explain why the problem is undecidable.

We will work with the counter machines introduced by Minsky \[17\]. These machines are also known as register machines or Minsky machines. A \emph{counter machine} consists of

- a finite number of registers $X_1, \ldots, X_n$
- a finite number of instructions $I_1, \ldots, I_m$.

The registers store natural numbers. The instructions tell the machine how to manipulate these numbers. The machine starts by executing the instruction $I_2$. Each register stores 0 when the execution starts (our counter machines do not take input). The instruction $I_1$ is the unique halting instruction. This instruction does not modify any register, and the machine halts if and only if this instruction is executed. Each of the the remaining instructions $I_2, \ldots, I_m$ will either be an \emph{increment instruction} or a \emph{decrement instruction}. An increment instruction $I_i$ is of the form $I_i : X_j, I_k$. The instruction tells the machine to increment the natural number stored in the register $X_j$ by one and then proceed with instruction $I_k$. A decrement instruction $I_i$ is of the form $I_i : X_j, I_k, I_\ell$. The instruction tells the machine to decrement the natural number stored in the register $X_j$ by one if this is possible, that is, if the number is strictly greater than zero. If it is possible to decrement the number, the machine will proceed with instruction $I_k$; if it is not possible, the machine will proceed with instruction $I_\ell$. We assume that we have $i \neq k$ in any increment instruction and $i \not\in \{k, \ell\}$ in any decrement instruction (this makes it easier to see that some of the arguments below indeed are correct).

It is well known that it is undecidable if a counter machine that starts with every register set to zero will terminate with every register set to zero. Kurtz & Simon
simulate counter machines by FracTran programs. A FracTran program $f$ is a finite sequence $q_1, \ldots, q_s$ of non-negative rational numbers. The transition function $F_f$ for the FracTran program $f$ maps the natural number $x$ to $q_i x$ where $i$ is the least $i$ such that $q_i x$ is integral. If no such $i$ exists, we regard $F_f(x)$ as undefined.

Following Kurtz & Simon [14], we will show how we given a counter machine $M$ can construct a FracTran program $f_M$ such that there exists $N$ such that $F^N_{f_M}(3) = 2$ iff $M$ terminates with every register set to zero. Let $p_i$ denote the $i$th prime ($p_1 = 2$).

The execution of a counter machine can be viewed as a sequence of configurations of the form

$$I_1, a_1, a_2, \ldots, a_n$$

where $I_i$ is the instruction to be executed next and $a_1, a_2, \ldots, a_n$, respectively, are the numbers stored in the registers $X_1, \ldots, X_n$. We represent a configuration by a natural number of the form

$$p_i p_{m+1}^{a_1} p_{m+2}^{a_2} \cdots p_{m+n}^{a_n}$$

where $i \leq m$. Let $f_M$ be a FracTran program $q_1, \ldots, q_s$ such that

1. For each increment instruction $I_i : X_j, I_k$ of $M$’s, there is $t$ such that $q_t = p_k p_{m+j}/p_i$.
2. For each decrement instruction $I_i : X_j, I_k, I_\ell$ of $M$’s, there is $t$ such that $q_t = p_k/p_i p_{m+j}$ and $q_{t+1} = p_\ell/p_i$.
3. $q_1, \ldots, q_s$ is a minimal sequence that satisfies (1) and (2) (so no sequence of length $s - 1$ will satisfy (1) and (2)).

Let us make some observations. Let $I_i$ be the increment instruction $I_i : X_j, I_k$. Then $p_k p_{m+j}/p_i$ is the one and only $q_t$ in the sequence $q_1, \ldots, q_s$ such that $q_t p_i p_{m+1}^{a_1} \cdots p_{m+n}^{a_n}$ is integral. Thus, we have

$$F_{f_M}(p_i p_{m+1}^{a_1} \cdots p_{m+j}^{a_j} \cdots p_{m+n}^{a_n}) = \frac{p_k p_{m+j}}{p_i} p_i p_{m+1}^{a_1} \cdots p_{m+j}^{a_j} \cdots p_{m+n}^{a_n}$$

$$= p_k p_{m+1}^{a_1} \cdots p_{m+j}^{a_j+1} \cdots p_{m+n}^{a_n}.$$
If $I_i$ is the decrement instruction $I_i : X_j, I_k, I_\ell$ and $X_j$ stores 0, then $p_\ell/p_i$ is the only $q_i$ in the sequence that makes $q_\ell p_i p_m^{a_1} \ldots p_{m+n}^{a_n}$ integral, and we have

$$F_M(p_i p_m^{a_1} \ldots p_{m+n}^{a_n}) = p_i p_m^{a_1} \ldots p_{m+n}^{a_n}.$$  

Any counter machine starts in the configuration $I_2, 0, 0, \ldots, 0$ (our counter machines do not take input and every register stores zero when the execution starts). This start configuration is represented by the prime $p_2$, that is, 3. If a counter machine halts with every register set to 0, it halts in the configuration $I_1, 0, 0, \ldots, 0$. This configuration is represented by the prime $p_1$, that is, 2. Given the observations above, it should be obvious that there exists $N$ such that $F_M^N(3) = 2$ iff $M$ halts with every register set to 0. Thus, we conclude that it is undecidable if there exists $N$ such that $F_M^N(3) = 2$.

A Collatz function is a function $C : \mathbb{N} \to \mathbb{N}$ that can be written of the form

$$C(x) = a_i x + b_i \quad \text{if } x \equiv i \mod M$$

where $a_i, b_i \in \mathbb{Q}$ and $M \in \mathbb{N}$ (for more on Collatz functions and the related Collatz problem, see [14]). Following the ideas of Kurtz & Simon [14], we will now construct a Collatz function $C_M$ of the more restricted form

$$C_M(x) = a_i x \quad \text{if } x \equiv i \mod M.$$  

such that we have $C_M(x) = F_M(x)$ whenever $F_M(x)$ is defined. We use the FracTran program $f_M = q_1, \ldots, q_s$ to determine $M$ and $a_0, a_1, \ldots, a_{M-1}$.

For each $q_i$ in the FracTran program $f_M = q_1, \ldots, q_s$, let $c_i, d_i \in \mathbb{N}$ be such that $c_i/d_i = q_i$ and $c_i$ and $d_i$ are relatively prime. Set $M$ to the least common multiple of $d_1, \ldots, d_s$ and then follow the procedure (I) to determine $a_0, a_1, \ldots, a_{M-1}$.

When the procedure starts every $a_j$ is undefined.

**PROCEDURE (I):**

for $i := 1, \ldots, s$

for $j := 0, \ldots, M - 1$

$$a_j := \begin{cases} a_j & \text{if } a_j \text{ already is defined} \\ q_i & \text{if } a_j \text{ is not yet defined and } d_i \text{ divides } j \\ \text{undefined} & \text{otherwise} \end{cases}.$$

This completes the construction of $C_M$. Many of the rationals $a_0, a_1, \ldots, a_{M-1}$ might still not be defined when the procedure terminates, but that is not important to us. The following claim is what matters to us:

$$F_M(x) = q_i x \iff C_M(x) = q_i x. \quad \text{(claim)}$$

In order to see that our claim holds, assume that $F_M(x) = q_i x$. Then, we know that $q_i x \in \mathbb{N}$ and that $q_i x \not\in \mathbb{N}$ when $\ell < i$. Let $x = zM + j$ where $j < M$ and let $c_i/d_i = q_i$ where $c_i$ and $d_i$ are relatively prime. Then, we have

$$q_i x = \frac{c_i}{d_i} (zM + j) \in \mathbb{N}. $$
As \( d_i \) divides \( M \) but not \( c_i \), it must be the case that \( d_i \) divides \( j \). A symmetric argument yields that \( d_i \) does not divide \( j \) when \( \ell < i \). Thus, our procedure sets \( a_j \) to \( q_i \). Hence, we have \( C_M(x) = C_M(zM + j) = a_jx = q_i x \). This proves that the left-right implication of the claim holds. It is easy to see that also the right-left implication holds.

It follows from the claim that \( f^N_{M}(3) = 2 \) iff \( C^N_M(3) = 2 \). Thus, we conclude that it is undecidable if there exists \( N \) such that \( C^N_M(3) = 2 \) (since we already have concluded that it is undecidable if there exists \( N \) such that \( f^N_{M}(3) = 2 \)).

We are soon ready to conclude that the Modulo Problem is undecidable. The function \( f \) in the Modulo Problem is a total function. Now, \( C_M \) is by no means a total function, but if \( C_M(x) \) is defined and \( x = zM + j \) where \( j < M \), then it turns out that we have \( A_j, B_j \in \mathbb{N} \) such that \( C_M(x) = A_jz + B_j \). To see that such \( A_j \) and \( B_j \) exist, observe that \( C_M(zM + j) = d \in \mathbb{N} \) for some relatively prime natural numbers \( c \) and \( d \). Moreover, the algorithm for constructing \( C_M \) assures that \( d \) divides both \( M \) and \( j \). Hence, let \( A_j = cM/d \) and let \( B_j = cj/d \), and \( A_j \) and \( B_j \) will be natural numbers such that \( C_M(x) = A_jz + B_j \). This entails that the following procedure will construct a sequence \( \langle A_0, B_0 \rangle, \ldots, \langle A_{M-1}, B_{M-1} \rangle \) of pair of natural numbers:

**PROCEDURE (II):**

for \( j := 0, \ldots, M - 1 \) do
- set \( A_j = cM/d \) and \( B_j = cj/d \) if procedure (I) defines \( a_j \) to equal \( c/d \)
- set \( A_j = B_j = 0 \) if procedure (I) leaves \( a_j \) undefined.

Now, let

\[
    f(x) = A_j z + B_j
\]

if there exists \( j < M \) such that \( x = zM + j \). Then, we obviously have \( f(x) = C_M(x) \) whenever \( C_M(x) \) is defined. Thus, as it is undecidable if there exists \( N \) such that \( C^N_M(3) = 2 \), it is also undecidable if there exists \( N \) such that \( f^N(3) = 2 \). These considerations should make it clear that the Modulo Problem is undecidable.

### 4.4 Undecidable Fragments

The stage is now set for our undecidability results. We know that the Modulo Problem is undecidable. Let \( \langle A_0, B_0 \rangle, \ldots, \langle A_{M-1}, B_{M-1} \rangle \) be an instance of the problem, and let

\[
    f(x) = A_j z + B_j
\]

if there exists \( j < M \) such that \( x = zM + j \). It is easy to see that there is \( N \) such that \( f^N(3) = 2 \) if and only if there exists a bit string of the form

\[
    01^{a_0+1}01^{a_1+1}0 \ldots 01^{a_N+1}0 \quad (*)
\]

where
First-Order Concatenation Theory with Bounded Quantifiers

- \( a_0 = 3 \) and \( a_N = 2 \)
- for each \( i \in \{0, \ldots, N-1\} \) there is \( z \in \mathbb{N} \) and \( j < M \) such that
  \[
  a_i = zM + j \quad \text{and} \quad a_{i+1} = A_jz + B_j .
  \]

The challenge is to claim the existence of a bit string \( x \) of the form (*) by using as few quantifiers as possible. We will of course need one unbounded existential quantifier to claim the existence of \( x \). Then we will need some quantifiers to state that \( x \) is of the desired form. In the structure \( \mathcal{D} \), we will state that any prefix of \( x \) of the form \( y_01 \) can be extended to a prefix of \( x \) of the form

\[
y_01 \overbrace{111 \ldots 111}^{kM \text{ copies}} \overbrace{01 \ldots 111}^{j \text{ copies}} \overbrace{01 \ldots 111}^{kA_j \text{ copies}} \overbrace{01 \ldots 111}^{B_j \text{ copies}} \overbrace{0}^{(*)}
\]

for some \( j \) and \( k \). Recall that \( j, M, A_j, B_j \) are fixed natural numbers. Thus, for each \( j \in \{0, \ldots, M-1\} \), we can express that \( x \) has a prefix of the form (**) by a formula \( \exists z [\psi_j(x, y, z)] \) where

\[
\psi_j(x, y, z) \equiv z1 = 1z \land y_01 \overbrace{zz \ldots zz}^{M \text{ copies}} \overbrace{111 \ldots 111}^{j \text{ copies}} \overbrace{01 \ldots 111}^{kA_j \text{ copies}} \overbrace{01 \ldots 111}^{B_j \text{ copies}} 0 \preceq x .
\]

In addition we have to state that \( x \) should start with \( 011110 \) and end with \( 01110 \). This explains why the formula

\[
(\exists x) \left[ 011110 \preceq x \land (\forall y \preceq x) \left[ y_01 \not\preceq x \lor y_01110 = x \right. \right.
\]

\[
\left. \quad \lor (\exists z) \left[ \bigvee_{j=0}^{M-1} \psi_j(x, y, z) \right] \right] \right]
\]

claims the existence of a bit string of the form (*). The formula contains two unbounded existential quantifiers and one bounded universal quantifier. If we could decide if such a formula is true in \( \mathcal{D} \), then we could decide the Modulo Problem. Hence, \( \Sigma_{D,2,0,1} \) is an undecidable fragment.

Similar reasoning will show that the fragments \( \Sigma_{B,1,0,1} \) and \( \Sigma_{F,1,3,2} \) are undecidable. The details can be found below.

**Theorem 40.** The fragments \( \Sigma_{2,0,1}^{D}, \Sigma_{1,0,1}^{B} \) and \( \Sigma_{1,3,2}^{F} \) are undecidable.

**Proof.** For any \( L_{BT} \)-term \( t \) let \( [t]^0 \equiv c \) and let \( [t]^{n+1} \equiv t \circ [t]^n \). Furthermore, note that the word equation \( x1 = 1x \) is satisfied iff \( x \in \{1\}^* \).

Let \( \langle A_0, B_0 \rangle, \ldots, \langle A_{M-1}, B_{M-1} \rangle \) be an instance of the Modulo Problem.

First we prove that \( \Sigma_{D,2,0,1} \) is undecidable. For \( i \in \{0, \ldots, M-1\} \), let

\[
\psi_j(x, y, z) \equiv z1 = 1z \land y_01[z]^M[1]^{j}01[z]^A_j[1]^B_j 0 \preceq x .
\]
and let \( \phi \) be the \( \Sigma_{2,0,1} \)-formula

\[
(\exists x) \left[ 01[1]^3 0 \preceq x \land (\forall y \preceq x)[ y01 \not\preceq x \lor y01[1]^2 0 = x \\
\lor (\exists z) \left[ \bigvee_{j=0}^{M-1} \psi_j(x, y, z) \right] \right]
\]

Then \( \mathcal{D} \models \phi \) if and only if the instance has a solution. Obviously, there is an algorithm for constructing \( \phi \) when the instance is given, and thus no algorithm can decide if a \( \Sigma_{2,0,1} \)-sentence is true in \( \mathcal{D} \), that is, the fragment \( \Sigma_{2,0,1}^\mathcal{D} \) is undecidable.

Next we prove that \( \Sigma_{1,0,1}^\mathcal{B} \) is undecidable. For \( j \in \{0, \ldots, M-1\} \), let \( \xi_j(y, x) \) be the formula

\[
\left( y1 = 1y \land 01[y]^M[1]^j 0 \subseteq x \land [y]^M[1]^j \not\equiv [1]^2 \right) \rightarrow 01[y]^M[1]^j 01[y]^A_j[1]^B_j 0 \subseteq x.
\]

Note that \( \xi_j(y, x) \) is trivially equivalent to a \( \Sigma_{0,0,0} \)-formula \( \check{\xi}_j(y, x) \). Let \( \phi' \) be the \( \Sigma_{1,0,1} \)-formula

\[
(\exists x) \left[ 01[1]^3 0 \subseteq x \land (\forall y \subseteq x)[ \bigvee_{j=0}^{M-1} \check{\xi}_j(y, x) ] \right].
\]

Then \( \mathcal{B} \models \phi' \) if and only if the instance has a solution. There is an algorithm for constructing \( \phi' \) from the instance, and thus we conclude that \( \Sigma_{1,0,1}^\mathcal{B} \) is undecidable.

We are left to prove that \( \Sigma_{3,1,2}^\mathcal{F} \) is undecidable. The following formula does to job:

\[
(\exists x) \left[ (\exists v \preceq x)[ 01[1]^3 0 v = x ] \land (\forall uv \preceq x)[ u01v \not\preceq x \lor \\
u01[1]^2 0 = x \land (\exists zy \preceq x)[ \bigvee_{j=0}^{M-1} \eta_j(z, u, y, x) ] ] \right]
\]

where

\[
\eta_j(z, u, y, x) = z1 = 1z \land u01[z]^M[1]^j 01[z]^A_j[1]^B_j 0 y = x
\]

for \( j \in \{0, \ldots, M-1\} \).

\[\square\]

**Corollary 41.** (i) It is undecidable whether a sentence of the form

\[
(\exists x_1)(\forall y \preceq x_1)(\exists x_2) \ldots (\exists x_n) s = t
\]
is true in $D$. (ii) It is undecidable whether a sentence of the form

$$(\exists x_1)(\forall y_1 y_2 \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n) s = t$$

is true in $B$. (iii) It is undecidable whether a sentence of the form

$$(\exists x_1)(\forall y_1 y_2 \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n) s = t$$

is true in $F$.

Proof. Consider the $\Sigma_{2,0,1}$-sentence $\phi$ if the proof of the preceding theorem. By Theorem 32, $\phi$ is $D$-equivalent to sentence of the form

$$(\exists x_1)(\forall y \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n) s = t.$$ 

This proves that (i) holds. Furthermore, we have

$$D \models (\exists x_1)(\forall y \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n) s = t$$

if and only if

$$B \models (\exists x_1)(\forall y \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n)[ y_1 \circ y_2 \neq x_1 \lor s = t ]$$

if and only if

$$F \models (\exists x_1)(\forall y \sqsubseteq x_1)(\exists x_2) \ldots (\exists x_n)[ y_1 \circ y_2 \neq x_1 \lor s = t ].$$

Thus, (ii) and (iii) hold by Lemma 29 and Lemma 30. \qed

References

1. B"uchi, J. R. and Senger, S.: Codi ng in the existential theory of concatenation. Archiv f"ur mathematische Logik und Grundlagenforschung 26 (1986/7), 101-106.
2. Corcoran, J., Frank, W. and Maloney, M.: String theory. The Journal of Symbolic Logic 39 (1974), 625-637.
3. Day, J., Ganesh, V., He, P., Manea, F. and Nowotka, D.: The satisfiability of extended word equations: The boundary between decidability and undecidability. arXiv:1802.00523 (2018).
4. Ganesh, V., Minnes, M., Solar-Lezama, A. and Rinard, M. C.: Word equations with length constraints: What is decidable? In: Biere A., Nahir A., Vos T. (eds), Hardware and Software: Verification and Testing, HVC 2012. LNCS 7857, pp. 209-226. Springer, Berlin, Heidelberg.
5. Grzegorczyk, A.: Undecidability without arithmetization. Studia Logica 79 (2005), 163-230.
6. Grzegorczyk, A. and Zdanowski, K.: Undecidability and concatenation. In: Ehrenfeucht et al. (eds), Andrzej Mostowski and Foundational Studies, pp. 72-91, IOS, Amsterdam, 2008.
7. Hajek, P. and Pudlak, P.: Metamathematics of first-order arithmetic. Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1993.
8. Halfon, S., Schnoebelen, P. and Zetzsche G: Decidability, complexity, and expressiveness of first-order logic over the subword ordering. In: 2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), DOI: 10.1109/LICS40289.2017

9. Hermes, H.: Semiotik. Eine Theorie der Zeichengestalten als Grundlage für Untersuchungen von formalisierten Sprachen. Vol. 5 of “Forschungen zur Logik und zur Grundlegung der exakten Wissenschaften”, Neue Folge, S. Hirzel, Leipzig, 1938.

10. Higuchi, K. and Horihata, Y.: Weak theories of concatenation and minimal essentially undecidable theories. Archive for Mathematical Logic 53 (2014), 835-853

11. Horihata, Y.: Weak theories of concatenation and arithmetic. Notre Dame Journal of Formal Logic 53 (2012), 203-222.

12. Karhumäki, J., Mignosi, F. and Plandowski, W.: The expressibility of languages and relations by word equations. Journal of the ACM 47 (2000), 483-505.

13. Kristiansen, L. and Murwanashyaka, J.: Decidable and undecidable fragments of first-order concatenation theory. In: Florin Manea, Russel G. Miller and Dirk Nowokta (eds), CiE 2018 – Sailing Routes in the World of Computation, LNCS 10936, pp. 244-253, Springer-Verlag 2018.

14. Kurtz S.A. and Simon J.: The undecidability of the generalized Collatz problem. In: Cai JY., Cooper S.B., Zhu H. (eds), TAMC 2007 – Theory and Applications of Models of Computation, LNCS 4484, pp. 542-553, Springer, Berlin, Heidelberg, 2007.

15. Leary, C. and Kristiansen, L.: A friendly introduction to mathematical logic. 2nd Edition, Milne Library, SUNY Geneseo, Geneseo, NY, 2015.

16. Makanin, G. S.: The problem of solvability of equations in a free semigroup. Mathematics of the USSR-Sbornik 32 (1977), 129-198.

17. Minsky, M. L.: Recursive unsolvability of Post’s problem of "tag" and other topics in theory of Turing machines. Annals of Mathematics 74 (1961), 437-455.

18. Post, E. L.: A variant of a recursively unsolvable problem. Bulletin of the American Mathematical Society 52 (1946), 264-268.

19. Quine, W. V.: Concatenation as a basis for arithmetic. Journal of Symbolic Logic 11 (1946), 105-114.

20. Quine, W. V.: Review of Hermes [9]. Journal of Symbolic Logic 4 (1939), 87-88.

21. Senger, S.: The existential theory of concatenation over a finite alphabet. PhD dissertation, Purdue University, 1982.

22. Svejdar, V.: On interpretability in the theory of concatenation. Notre Dame Journal of Formal Logic 50 (2009), 87-95.

23. Tarski, A.: Der Wahrheitsbegriff in den formalisierten Sprachen. Studia Philosophica 1 (1935), 261-405.

24. Visser, A.: Growing commas. A study of sequentiality and concatenation. Notre Dame Journal of Formal Logic 50 (2009), 61-85.