Decidability problems in languages with Henkin quantifiers*

Michał Krynicki
Institute of Mathematics, Warsaw University, Warsaw, Poland

Marcin Mostowski
Department of Logic, Philosophical Faculty, Warsaw University, Warsaw, Poland

Communicated by A. Lachlan
Received 14 May 1990
Revised 12 September 1991

Abstract


We consider the language $L(H_n)$ with all Henkin quantifiers $H_n$ defined as follows: $H_n(x_1, \ldots, x_n, y_1, \ldots, y_n) \iff \exists f_1, \ldots, f_n \forall x_1, \ldots, x_n \phi(x_1, \ldots, x_n, f_1(x_1), \ldots, f_n(x_n))$. We show that the theory of equality in $L(H_n)$ is undecidable. The proof of this result goes by interpretation of the word problem for semigroups.

Henkin quantifiers are strictly related to the function quantifiers $F_n$ defined as follows: $F_n(x_1, \ldots, x_n, y_1, \ldots, y_n) \iff \exists f_1, \ldots, f_n \forall x_1, \ldots, x_n \phi(x_1, \ldots, x_n, f_1(x_1), \ldots, f_n(x_n))$. In contrast with the first result we show that the theory of equality with all quantifiers $F_n$ is decidable.

We also consider decidability problems for other theories in languages $L(F_n)$ and $L(H_n)$.

1. Introduction

1.1. Introductory remarks

The quantifier is one of the crucial concepts in modern logic. However, for a long time the only quantifiers considered were the existential ($\exists$) and universal ($\forall$) ones. The first general concept of a quantifier was proposed by Andrzej Mostowski, later generalized by P. Lindström. Both definitions accept as quantifiers a lot of concepts which we are accustomed to treat in a very different way.

Correspondence to: M. Krynicki, Instytut Matematyki, Uniwersytet Warszawski, Banacha 2, 02-097 Warszawa, Poland. e-mail: krynicki@plearn.

* Research reported here has been supported by Polish Government Grant RPBP.III.24, B4.

0168-0072/92/$05.00 © 1992 — Elsevier Science Publishers B.V. All rights reserved
way. Probably one of the most natural concepts of quantifier not covered by traditional theory was suggested by L. Henkin.

Henkin [6] proposed to consider a language with quantifiers arising from existential and universal quantifiers by writing them down in some orderings (i.e., dense ordering, partial ordering etc.). He called them dependent quantifiers. As simplest example, he considered a quantifier H bounding four variables in a formula. A formula $\phi$ built up using H can be written as:

$$\forall x \exists z \varphi(x, y, z, u).$$

Its semantical meaning can be explained as $\exists f, g \forall x, y \varphi(x, y, fx, gy)$. It was proved (see Henkin [6]) by Ehrenfeucht that the logic $L(H)$ is essentially stronger than first-order logic and the set of tautologies of $L(H)$ is not recursively enumerable. About ten years later Enderton [5] and Walkoe [16] considered logic with quantifiers proposed by Henkin but bounding finitely many variables only. They call them ‘finitely partially ordered quantifiers’ (f.p.o.q.). Instead of this name we shall use the term ‘Henkin quantifiers’. In the literature also the names ‘branched quantifiers’ or, in a little broader sense, ‘branching quantifiers’ are used.

After the two eminent results of Gödel that elementary logic is recursively enumerable and that no theory rich enough to contain arithmetic could be recursively enumerable, and Church’s theorem that elementary logic is not decidable, we have two important borderlines dividing logics and theories into three classes:

1. decidable,
2. undecidable but recursively axiomatizable,
3. not recursively axiomatizable.

Traditionally the majority of efforts were devoted to identifying the borderline between decidable and undecidable. However, the second borderline seems also to be very important. A recursively enumerable theory can be axiomatized in a standard way, hence its set of theorems can be defined by proof-theoretical means only, not referring to any semantical description.

The main purpose of this paper is to draw as accurate as possible the borderline between decidable and undecidable theories in languages with Henkin quantifiers.

We state two theorems contrasting undecidability of some simple fragment of logic with empty signature in a language with Henkin quantifiers on the one hand, with decidability of a similar fragment with function quantifiers, being apparently an inessential weakening of Henkin quantifiers, on the other hand.

Researches and results presented in this paper are strictly related to those from Krynicki–Lachlan [9], where some rough borderlines between decidable and undecidable fragments of $L(H)$ were drawn. Here we give a more detailed description of this borderline, from two sides—positive and negative—of decidability cases.
The main part of this paper is divided into two parts (Sections 2 and 3). In Section 2 the theory of equality is discussed. Section 3 is devoted to investigations of decidability problems for some theories of nonempty signature.

1.2. Basic concepts

Formally a Henkin quantifier prefix can be defined as $Q = (A_Q, E_Q, D_Q)$, where $A_Q$ and $E_Q$ are disjoint sets of variables (universal and existential, respectively), and $D_Q \subseteq A_Q \times E_Q$ ($D_Q$ is a dependency relation, it says on which universal variables a given existential variable depends). $Q$ bounds those variables which belong to $A_Q \cup E_Q$. We differentiate between quantifiers and quantifier prefixes as follows: a quantifier prefix is a quantifier with assigned concrete variables. For instance $\forall$ is a quantifier, but $\forall x$ is a quantifier prefix. In this sense we have defined Henkin quantifier prefixes, not Henkin quantifiers. However, this definition can be modified in a natural way to apply it to Henkin quantifiers also. We will use a somewhat ambiguous notation and terminology not differentiating between quantifiers and quantifier prefixes if it will be safe enough.

Semantics for Henkin quantifiers is given through translation into second-order logic. A formula $Q \phi(x_1, \ldots, x_n, y_1, \ldots, y_m)$, where $x_1, \ldots, x_n$ are all universal variables of $Q$ and $y_1, \ldots, y_m$ are all existential variables from $Q$, is translated into a formula

$$\exists f_1 \cdots \exists f_m \forall x_1 \cdots \forall x_n \exists (\phi(x_1, \ldots, x_n, f_1(\vec{x}_1), \ldots, f_m(\vec{x}_m)))$$

where $\vec{x}_i$ is a sequence of variables being in relation $D_Q$ with $y_i$ (or in other words on which $y_i$ depends). In what follows we will use a more intuitive notation introduced by Henkin. Instead of $\{x, y\}, \{z, u\}, \{(x, z), (y, u)\}$ we shall write

$$\forall x \exists z$$
$$\forall y \exists u$$

Even if not all Henkin quantifiers can be written in this more intuitive way, this is no problem since we shall consider here only quantifiers having such intuitive representations.

We assume the following notation: $H^k_n$ is the quantifier

$$\forall x_{11} \cdots \forall x_{1k} \exists y_1$$
$$\vdots$$
$$\vdots$$

$$\forall x_{n1} \cdots \forall x_{nk} \exists y_n$$

By default $k = 1$, $n = 2$—which means that $H_n$ is $H^1_n$, and $H^k$ is $H^2_n$, and in particular $H$ is $H^1_1$ (it is so-called the Henkin quantifier). By $L(H^k_n)$ we mean the logic with $H^k_n$ as the only quantifier. $L(H_n)$ is the logic with all quantifiers $H_n$. By $L^*$ we denote the logic with all Henkin quantifiers.

In the paper by Walkoe [16] two important normal form characterizations for Henkin quantifiers were shown. He proved:
Theorem 1. For every Henkin quantifier prefix $Q$ and every formula $\phi$ there are effectively found quantifier prefixes $Q'$ and $Q''$ and quantifier-free formulae $\alpha, \beta, \alpha', \beta'$, with identity as the only predicate such that $Q\phi$ and $Q'(\alpha \land (\beta' \Rightarrow \phi))$ are equivalent, and $Q\phi$ and $Q''(\alpha' \land (\beta' \Rightarrow \phi))$ are equivalent in infinite models, where $Q'$ is of the form
\[ \forall x_1 \cdots \forall x_k \exists y_1 \cdots \exists y_n \]
\[ \forall x_1' \cdots \forall x_k' \exists y_1' \cdots \exists y_n' \]
and $Q''$ is of the form
\[ \forall x_1 \forall x_1' \exists y_1 \]
\[ \vdots \]
\[ \forall x_n \forall x_n' \exists y_n \]

It follows that using quantifiers of the form $Q'$ we can define all Henkin quantifier prefixes.

It is not known whether prefixes of the form $Q''$ are sufficient without the restriction to infinite models, or if the semantical power of prefixes of the form $H_n$ is sufficient.

The logic with Henkin quantifiers is stronger than that of elementary logic. The first known sentence with a Henkin quantifier not expressible in elementary logic is the so-called Ehrenfeucht statement:
\[ \neg \exists t \forall y \exists u ((x = y \Leftrightarrow z = u) \land t \neq z). \]

It says that the universe is finite. Translated into second-order logic it says exactly what follows: it is not so that there is a one-one function and an object which is different from all values of this function. The equivalence $(x = y \Leftrightarrow z = u)$ is used here to express two things: that two functions are identical (implication from left to right), and that this function is one-one (implication from right to left). It follows that the Ehrenfeucht statement can be expressed using a weaker quantifier, a so-called function quantifier. Function quantifiers were introduced in Krynicki [8] and studied in Krynicki–Väänänen [10]. The function quantifier $F_n$ binds $2n$ variables in one formula and its semantics is defined as follows
\[ (F_n x_1 \cdots x_n y_1 \cdots y_n) \phi(x_1, \ldots, x_n, y_1, \ldots, y_n) \]
\[ \text{iff } \exists f \forall x_1 \cdots x_n \phi(x_1, \ldots, x_n, f(x_1), \ldots, f(x_n)). \]

It is known that $L(F_n)$ is a sublogic of $L(H_n)$. $(F_n x_1 \cdots x_n y_1 \cdots y_n) \phi(x_1, \ldots, x_n, y_1, \ldots, y_n)$ is equivalent to $(H_n x_1 \cdots x_n y_1 \cdots y_n) (\alpha \land \phi(x_1, \ldots, x_n, y_1, \ldots, y_n))$, where $\alpha$ is a conjunction of formulae $(x_i = y_i \Rightarrow y_i = y_i)$ for $i = 2, \ldots, n$. By $L(F_n)$ we denote the logic with all quantifiers $F_n$. The concept of function quantifier can be generalized in a natural way. So $F_n^k$ is a function quantifier, where $n$ is the number of tokens of the function used, and $k$ is its arity ($F_n^k$ binds...
Decidability problems in languages with Henkin quantifiers

$k(n + 1)$ variables. Thus for instance

\[(F^2_2 x_1 y_1, y_2 z_1, z_2) \phi(x_1, x_2, y_1, y_2, z_1, z_2)\]

is equivalent to

\[\exists f \forall x_1 x_2 y_1 y_2 \phi(x_1, x_2, y_1, y_2, f(x_1, y_1), f(x_2, y_2)).\]

To compare logics we use the relation \(\leq\) defined as follows: \(L \leq L'\) if and only if for every \(L\)-formula \(\phi\) there is an \(L'\)-formula \(\psi\) such that for every structure of proper signature \(\mathcal{A}: \mathcal{A} \vDash \phi\) if and only if \(\mathcal{A} \vDash \psi\). \(L \equiv L'\) if and only if \(L \leq L'\) and \(L' \leq L\).

We follow more or less the standard notation, as in Chang–Keisler [2]. By \(f^n(a)\) we denote the result of \(n\)-fold application of \(f\) to \(a\) (i.e., \(f^0(a) = a, f^{n+1}(a) = f(f^n(a))\)).

2. Theory of equality

2.1. First-order formulation of the word problem

Let \(A^*\) be the set of words over the alphabet \(A = \{a, b\}\). The word problem for semigroups can be formulated as the decision problem for sentences of the form

\[a_1 = b_1 \land \cdots \land a_n = b_n \Rightarrow a_{n+1} = b_{n+1},\]

where \(a_i, b_i \in A^*\) for \(i = 1, 2, \ldots, n + 1\). That is, as the problem of deciding if a sentence of the form (1) is true in every semigroup with two generators \(a, b\). It is well known that this problem is undecidable. In Davis [3] this problem is formulated in a slightly more general way. The difference is that he allows any finite alphabet \(A\). However, from an algorithmic point of view both formulations are equivalent.

Let \(L\) be a first-order logic with identity and two unary function symbols \(f, g\) as only nonlogical constants. Let \(T[x]\) be the set of all terms of logic \(L\) with \(x\) as unique free variable. Let \(S\) be the set of all sentences of the form:

\[\forall x (s_1(x) = t_1(x)) \land \cdots \land \forall x (s_n(x) = t_n(x)) \Rightarrow \forall x (s_{n+1}(x) = t_{n+1}(x)),\]

where \(s_i(x), t_i(x) \in T[x]\) for \(i = 1, 2, \ldots, n + 1\).

By the tautology problem for the set \(S\) we mean the problem of deciding if a given sentence belonging to the set \(S\) is true in every model for \(L\).

**Lemma 1.** The tautology problem for the set \(S\) is undecidable.

**Proof.** We shall show that the word problem can be recursively reduced to the decision problem for the set \(S\).

Let us observe that there is a one-one effective correspondence \(\mu\) between the set \(A^*\) and \(T[x]\) such that \(\mu(\epsilon) = x (\epsilon\) is the empty word) and for \(a \in A^*\),
\(\mu(a\alpha) = f(\mu(\alpha))\) and \(\mu(b\alpha) = g(\mu(\alpha))\). Each model \(\mathfrak{A} = (X, F, G)\) for \(L\) determines a semigroup \(H\) of mappings from \(X\) to \(X\) generated by \(F\) and \(G\). Moreover, for any semigroup \(H\) with two generators there is a model \(\mathfrak{A}\) for \(L\) such that for every \(\alpha, \beta \in A^*\) we have:

\[\alpha = \beta \text{ is true in } H \text{ if and only if } \mathfrak{A} \models \forall x (\mu(\alpha) = \mu(\beta)).\]

Then the implication (1) is true in every semigroup with two generators exactly when the sentence:

\[\forall x (\mu(\alpha_1) = \mu(\beta_1)) \land \cdots \land \forall x (\mu(\alpha_n) = \mu(\beta_n)) \Rightarrow \forall x (\mu(\alpha_{n+1}) = \mu(\beta_{n+1}))\]

is true in every model for \(L\). □

Let \(W\) be the set of all sentences of the logic \(L\) of the form \(\forall x \exists y_1 \cdots \exists y_n \phi\), where \(\phi\) is a boolean combination of formulae of the form \(f(z) = v\) and \(g(z) = v\), where \(z, v \in \{x, y_1, \ldots, y_n\}\).

**Lemma 2.** The tautology problem for \(W\) is undecidable.

**Proof.** We are going to reduce the decision problem for \(S\) to that of \(W\). Let us observe that the sentence (2) is equivalent to a sentence of the form \(\forall x \exists y \phi\), where \(\phi\) is a quantifier-free formula. Then we can eliminate all complex terms from any quantifier-free formula \(\eta\) subsequently replacing \(\eta(f(s))\) by \(\exists z (z = s \land \eta(f(z)))\), where \(z\) is a new variable not occurring in \(\eta\). In this way we effectively obtain a formula belonging to \(W\) and equivalent to (2). □

### 2.2. Undecidability of the theory of equality in \(L(H_m)\)

Let \(\phi\) be a sentence of the form \(\forall z_0 \exists z_1 \cdots \exists z_n \psi\), where \(\psi\) is a boolean combination of formulae of the form \(f(z) = v\) and \(g(z) = v\), where \(z, v \in \{z_0, z_1, \ldots, z_n\}\) (it means \(\phi \in W\)). By \(\phi^*\) we denote the following sentence

\[
\forall x_1 \exists y_1 \\
\vdots \\
\exists z_0 \forall x_m \exists y_m \forall z_1 \cdots \forall z_n (\xi \land \xi \land (\chi \Rightarrow \psi')), \\
\forall u_1 \exists v_1 \\
\vdots \\
\forall u_k \exists v_k
\]

where \(\xi\) is the conjunction \((x_1 = x_2 \Rightarrow y_1 = y_2) \land \cdots \land (x_1 = x_m \Rightarrow y_1 = y_m)\), \(\xi\) is the conjunction \((u_1 = u_2 \Rightarrow v_1 = v_2) \land \cdots \land (u_1 = u_k \Rightarrow v_1 = v_k)\), \(\psi'\) is the negation of the formula obtained from \(\psi\) by substitutions \(y_i\) in place of the \(i\)th occurrence of \(f\) in \(\psi\), \(v_i\) in place of the \(i\)th occurrence of \(g\) in \(\psi\), \(\chi\) is the conjunction of equalities \(x_i = z_j\) and \(u_i = z_j\), when the \(i\)th occurrence of \(f\) in \(\psi\) is in a context \(f(z_j)\), and the \(i\)th occurrence of \(g\) in \(\psi\) is in a context \(g(z_j)\) (of course \(m\) and \(k\) are the numbers of occurrences in \(\psi\) of \(f\) and \(g\), respectively).
The described construction is very similar to that of Walkoe [16] and Enderton [5]. The point here is that to formulate \( \psi' \) we do not need all Henkin quantifiers but only the quantifiers \( H_n \).

A semantical analysis of sentences \( \phi \) and \( \phi^* \) proves the following.

**Lemma 3.** For every \( \phi \in W \), \( \vdash \phi \) if and only if \( \neg \phi^* \) is universally valid.

Let \( V \) be the set of sentences of the form \( \neg \mathcal{Q}\phi \), where \( \mathcal{Q} \) is a Henkin quantifier prefix and \( \phi \) is a purely logical (i.e., without nonlogical constants) quantifier-free formula. \( V_0 \) is the subset of formulae \( \neg \mathcal{Q}\phi \) from \( V \) such that \( \mathcal{Q} \) is \( H_n \) for some \( n \).

**Lemma 4.** The tautology problem for the set \( V \) is recursively enumerable.

**Proof.** Let \( \zeta \in V \). This means that \( \zeta \) is of the form \( \neg \mathcal{Q}\phi \), where \( \phi \) is quantifier-free. Then we can translate \( \zeta \) into a second-order statement in the usual way. Let \( \forall f_1 \cdots \forall f_n \psi \) be the second-order translation of \( \neg \mathcal{Q}\phi \). According to our assumption \( \psi \) can be identified with a first-order formula when we treat the second-order variables \( f_1, \ldots, f_n \) as nonlogical constants. Then we see that \( \zeta \) is a tautology if and only if \( \psi \) is a first-order tautology. Then we obtain the claim by the recursive enumerability of the set of first-order tautologies. \( \square \)

**Theorem 2.** The set of tautologies belonging to \( V_0 \) is recursively enumerable but not recursive, and consequently the theory of equality in \( L(H_n) \) is undecidable.

**Proof.** Recursive enumerability follows from Lemma 4. By Lemmas 2 and 3 the tautology problem for the set \( V_0 \) cannot be decidable. \( \square \)

It does not follow that the theory of equality in \( L(H_n) \) is axiomatizable (i.e., the set of its theorems is recursively enumerable). Let us notice that the implication (1) does not hold in all semigroups if and only if there is a semigroup of functions over an infinite set with two generators in which the implication (1) does not hold. Thus we have justified the following:

**Lemma 5.** Let \( \varphi \in S \). Then \( \neg \varphi \) is consistent if and only if \( \neg \varphi \) has an infinite model (of any infinite power by the Skolem–Löwenheim Theorem).

Let \( W^- \) be the set of negations of formulae from \( W \), that is, the set of sentences from logic \( L \) of the form \( \exists x \forall y_1 \cdots \forall y_n \phi \), where \( \phi \) is a boolean combination of formulae of the form \( f(z) = v \) and \( g(z) = v \), where \( z, v \in \{x, y_1, \ldots, y_n\} \).

**Lemma 6.** The problem whether \( \varphi \in W^- \) has an infinite model is not recursively enumerable.
Proof. The reasoning is similar to that in the proof of Lemma 2. We use here the fact that the problem whether implication (1) does not hold in all semigroups is not recursively enumerable, being the complement of a recursively enumerable but not recursive problem. \(\square\)

Let \(\phi \in W^\sim\). We define \(\hat{\phi}\) as the sentence \((y \Rightarrow \phi^*)\), where \(\phi^*\) is defined by (3) and \(\gamma\) is Ehrenfeucht’s statement saying that there are finitely many entities. By semantical analysis of \(\hat{\phi}\) we obtain the following:

Lemma 7. For \(\phi \in W^\sim\) we have: \(\phi\) has an infinite model if and only if \(\hat{\phi}\) is a tautology.

Now we get the conclusion.

Theorem 3. The theory of equality in \(L(H_\omega)\) is neither decidable nor axiomatizable.

Proof. By Lemmas 6, 7, the considered set has a recursively defined subset which is not recursively enumerable. By Theorem 2 it also has a recursively defined subset which is recursively enumerable but not recursive. \(\square\)

2.3. Bifunctional quantifiers

In the above reasoning justifying the last theorem we have used a fragment of the logic \(L(H_\omega)\). This suggests us to introduce the quantifier \(F_{m,n}\) defined as follows

\[
F_{m,n}x_1 \cdots x_m y_1 \cdots y_n \phi(x_1, \ldots, y_n) \quad \text{iff} \quad \exists f \forall x_1 \cdots x_m y_1 \cdots y_n \phi(x_1, \ldots, x_m, f(x_1), \ldots, f(x_m), y_1, \ldots, y_n, g(y_1), \ldots, g(y_n)).
\]

We call this quantifier bifunctional quantifier. The logic with all bifunctional quantifiers will be denoted by \(L(F_{\omega,\omega})\). The following gives us the relations which hold between bifunctional and other quantifiers.

Proposition. (a) If \(m \leq k\) and \(n \leq l\) or \(n \leq k\) and \(m \leq l\) then \(L(F_{m,n}) \leq L(F_{k,l})\).
(b) \(L(F_{m,n}) = L(F_{n,m})\).
(c) \(L(F_{1,1}) = L(H)\).
(d) \(L(F_{m,n}) \leq L(H_{m+n})\).
(e) \(L(F_{m+n}) = L(F_{m+n,0}) \leq L(F_{m,n})\).
(f) \(L(F_\omega) \leq L(F_{\omega,\omega}) \leq L(H_\omega)\).

As a corollary of our considerations we have:

Theorem 4. The theory of equality in \(L(F_{\omega,\omega})\) is neither decidable nor axiomatizable.
2.4. The proof system $LB$

In Mostowski [12] the proof system $LB$ for logic with Henkin quantifiers is defined by two rules in natural deduction style $LB1$, $LB2$. It was proved that $LB$ is complete with respect to the so-called weak semantics, defined by the class of structures $(\mathcal{A}, K)$, where $\mathcal{A}$ is a first-order structure, and $K$ is a class of relations over $|\mathcal{A}|$ closed under definability in $(\mathcal{A}, K)$ by $LB$-formulae. Satisfiability is given through a translation $t$ of $LB$-formulae into second-order formulae restricting second-order quantifiers to $K$. Thus

$$t\left(\begin{array}{c} Q' \\ Q \end{array}\right)(\varphi)$$

is defined as $\exists R, S \in K (t(QR(x)) \land t(Q'S(y)) \land \forall \bar{x}, \bar{y} (R(\bar{x}) \land S(\bar{y}) \Rightarrow t(\varphi)))$, where $\bar{x}$ and $\bar{y}$ are sequences of all variables occurring in $Q$ and $Q'$, respectively. Assuming the Axiom of Choice we obtain by removing the restriction to a class $K$, a semantics equivalent to that given by Henkin and assumed elsewhere in this paper.

Of course the set of $LB$-tautologies is recursively enumerable, as is true of all finitary proof systems. However, we can justify an analogy of Theorem 2 for $LB$. For this we need the following:

**Lemma 8.** For every $\varphi \in V$, $\varphi$ is a tautology if and only if $\varphi$ is $LB$-provable.

In Mostowski [14] there was proved a more general theorem for $\varphi$ having only negative occurrences of quantifier prefixes. The idea of that proof is similar to that used in the proof of Lemma 4.

By Theorem 2 and Lemma 8 we obtain the following:

**Theorem 5.** The set of $LB$-tautologies belonging to $V_0$ is not recursive.

2.5. A logic with one Henkin quantifier

In Mostowski [13] the strongest known negative result concerning the decidability of the theory of equality in a language with fixed Henkin quantifier is stated. Unfortunately the formulation of the last theorem in that paper stating the relevant fact is incorrect. The correct formulation is as follows.

**Theorem 6.** Let $T$ be a self-consistent (i.e., $\neg\text{CONS}(T)$ is not provable in $T$), finitely axiomatizable extension of Gödel–Bernays set theory. Then there is a Henkin quantifier $Q$ such that for every algorithm $A$ the set-theoretical statement expressing that “$A$ is a decision algorithm for $L(Q)$” is not provable in $T$.

The proof in Mostowski [13] gives a correct justification for this theorem. Furthermore the assumption about finite axiomatizability is not essential.
2.6. Decidability of the theory of equality in $L(F_n)$

In this section, in spite of the main result of Section 2.3, we are going to show that the theory of equality is decidable in the language $L(F_n)$. This result should be less surprising if we consider the following example. In Section 2.2 we have reduced the word problem to the tautology problem of sentences of the form $\neg H_n \phi$ where $\phi$ is a first-order formula. Then we have concluded the undecidability of this problem. Now let us consider the class of sentences of the form $\neg F_n \bar{x} \phi$ where $\phi$ is a first-order formula. It follows from the Ehrenfeucht theorem [4] concerning decidability of the first-order theory of one unary function that the tautology problem for the last class of sentences is decidable.

Now we start to prove our main result in this section. The proof will be done by the method of elimination of quantifiers, similar to that used in Krynicki–Lachlan [9]. Thus we will consider a formula $F_n x_1 \cdots x_n \phi$, where $\phi$ is a quantifier-free formula and we shall show that it can be equivalently replaced by some quantifier-free formula using some, so-called, basic sentences. To do this we shall classify all possible unary functions with respect to their properties expressed by quantifier-free formulas. Thus we will consider structure of the form

$$\mathfrak{A} = (A, a_1, \ldots, a_m, f),$$

where $a_1, \ldots, a_m \in A$, $a_i \neq a_j$ for $i \neq j$ and $f$ is a function from $A$ to $A$. We begin with defining a type of $n$ elements of such a model. It describes all equality relations between given elements and their images in the function $f$. Next we will introduce the notion of 1-basic type which describes possible neighbours of a given element in the graph of a function. This allows us to define the notion of code. A code is in some sense the 'skeleton' of a function—it contains all possible configurations of the graph of a given function. We will show that there are finitely many codes and that each function corresponds to some code. This allows us to consider only finitely many sorts of functions and to effectively determine these sorts of functions which satisfy a starting formula $\phi$. Finally, it suffices to associate with each code a suitable basic sentence which guarantees the existence of a function of a given sort.

By a type of elements $b_1, \ldots, b_n \in A$ in a structure $\mathfrak{A}$, denoted by $t^\mathfrak{A}(b_1, \ldots, b_n)$, we mean a pair $(\epsilon, \eta)$ where $\epsilon$ is a function from the set \{$(i, j): 0 < i < j \leq 2n$\} into \{0, 1\} such that

$$\epsilon(i, j) = 0 \quad \text{if and only if} \quad b_i = b_j,$$

and $\eta$ is a function from the set \{1, \ldots, 2n\} into \{0, \ldots, m\} such that

$$\eta(i) = \begin{cases} r & \text{if } b_i \neq a_r, \\ 0 & \text{if } b_i = a_r \text{ for } r = 1, \ldots, m, \end{cases}$$

where for $i = 1, \ldots, n$ we put $b_{n+i} = f(b_i)$.

Thus the type $t^\mathfrak{A}(b_1, \ldots, b_n)$ describes the graph of the function $f$ restricted to the elements $b_1, \ldots, b_n, f(b_1), \ldots, f(b_n)$. Moreover, this description is com-
Decidability problems in languages with Henkin quantifiers

1.59 complete. It means that if two graphs are not isomorphic then their types are not the same (however, not conversely). Thus with each type \( t \) we can associate a unique, up to isomorphism, directed graph \( G_t \). Moreover, for each type \( t \) a quantifier-free formula \( \phi_t(x_1, \ldots, x_n) \) can effectively be found such that for \( b_1, \ldots, b_n \in A \), \( \phi_t(b_1, \ldots, b_n) \) is true in \( \mathfrak{A} \) iff \( t^\mathfrak{A}(b_1, \ldots, b_n) = t \).

A type \( t \) is called connected if \( G_t \) is a connected graph (i.e., for all elements \( a, b \) of the graph there is a sequence \( a_1, \ldots, a_s \) such that \( a_1 = a \), \( a_s = b \) and for all \( i = 1, \ldots, s - 1 \) there is an arrow from \( a_i \) to \( a_{i+1} \) or from \( a_{i+1} \) to \( a_i \)). A type \( t = (\varepsilon, \eta) \) is called simple if for all \( i, j \) such that \( 0 < i < j < 2n \) we have:

\[
\varepsilon(i, j) = 0 \iff j = n + i - 1 \land i > 1
\]

and \( \eta(i) = 0 \) for \( i = 1, \ldots, n \) (this describes the situation where \( b_{i+1} = f(b_i) \) and \( b_i \neq b_j \) for \( i \neq j \)). A type \( t = (\varepsilon, \eta) \) is called ramifying if for some \( 1 \leq i < j \leq n \), \( \varepsilon(1, n + i) = \varepsilon(1, n + j) = 0 \) and \( \varepsilon(i, j) = 1 \). A type is called ramifying in the wider sense if in the graph \( G_t \) there are distinct arrows having the same endpoints. A type \( t = (\varepsilon, \eta) \) is called initial if for all \( i = 1, \ldots, n \), \( \varepsilon(1, n + i) = 1 \).

By \( t(n, m) \) we denote the number of all types. In an obvious way we have:

\[
t(n, m) \leq 2^{(2n-1)m + 2n}.
\]

Finally we define

\[
\text{TYP}^\mathfrak{A}(\mathbb{A}) = \{ t^\mathfrak{A}(b_1, \ldots, b_n) : b_1, \ldots, b_n \in A \}.
\]

Let \( D \) be a set. For \( d \in D \) let \( d \) be the name of \( d \). A basic formula on \( D \) is any formula of one of the following forms:

\[
\begin{align*}
f^*(x) &= f^*(x) \quad \text{for } 0 \leq l < s \leq n, \\
f^*(x) &= f^*(d) \quad \text{for } 0 \leq s + l \leq n, \\
\exists y_2 \cdots y_n \phi(x, y_2, \ldots, y_n) & \quad \text{for all connected } t.
\end{align*}
\]

The last sentence we will denote by \( \theta_t \).

A basic 1-type on \( D \) is a set \( p \) of basic formulas on \( D \) or their negations such that:

(i) for each basic formula \( \phi \) on \( D \) either \( \phi \) or \( \neg \phi \) is in \( p \);

(ii) \( p \) is realized in some structure of the form (4) in such a way that \( \{a_1, \ldots, a_m\} \subseteq D \subseteq A \).

A basic 1-type on \( D \) containing a formula \( "x = d" \) is called trivial. A basic 1-type containing a formula \( "f^*(x) = d" \) for some \( s \leq n \) and \( d \in D \) is called subtrivial. A basic 1-type containing for some \( s \leq n \) a formula \( "f^*(x) = x" \) is called cycle. A cycle is called \( r \)-cycle if \( r \) is the least number \( s \) such that the basic formula \( "f^*(x) = x" \) belongs to it. A trivial 1-type \( p \) is called exciting if for all \( d \in D \), \( "f(x) = d" \notin p \). A 1-type \( p \) is called simple if for some simple type \( t, \theta_t \in p \). Finally, a basic 1-type \( p \) is called semitrivial if for some \( d \in D \) and \( k \leq n \), \( "x = f^k(d)" \in p \).
Let $T(n, m)$ denote the number of basic 1-types on \{a_1, \ldots, a_n\}. We have $T(n, m) \leq 2^{s(n, m)}$ (observe that the occurrence of the formulas $f'(x) = f'(d')$ and $f''(x) = f''(y')$ in a basic 1-type and the occurrence of formulas $\theta_i$ in it are dependent). Let us denote

\begin{align*}
M'(n, m) &= 2n T(n, m)((n - 1)t(n, m) + 2n) \quad \text{and} \\
M(n, m) &= \max(M'(n, m), (n - 1)^n(n + m) + 1).
\end{align*}

By a code we mean a $m + 5$-tuple $(D, d_1, \ldots, d_m, P, Q, F, f^*)$ which satisfies the following conditions:

0. $\text{card}(D) \leq M(n, m)$.

1. $d_1, \ldots, d_m$ are pairwise distinct elements of $D$.

2. $P$ is a set of basic 1-types on $D$ such that the following conditions are satisfied:

\begin{itemize}
  \item for each $d \in D$ and $l \leq n$ there is $p \in P$ which contains a formula "$x = f'(d')$";
  \item if $p \in P$ is subtrivial but not trivial, then for some $d \in D$ "$f(x) = d'$" $\in p$;
  \item if $p \in P$ is not trivial and $t$ is ramifying, then $\theta_i \notin p$;
  \item if $p \in P$ is exciting, then the following conditions are satisfied:
    \begin{itemize}
      \item for simple type $t$, $\theta_i \in p$,
      \item for all initial and ramifying in wider sense types $t$, $\theta_i \notin p$,
      \item for every $d \in D$ and $k = 1, \ldots, n$, "$f^k(x) = d'$" $\notin p$;
    \end{itemize}
  \item if $p \in P$ is not trivial and for some type $t$, $\theta_i \in p$, then there is a trivial 1-type $p'$ such that $\theta_i \in p'$;
  \item if for some $r = 1, \ldots, n$ there is an $r$-cycle in $P$ which is not trivial, then the set \{\text{there is a trivial type $p$ such that "$x = d'$" $\in p$ is an $r$-cycle}\} has at least $n$ elements;
  \item the set of sentences \{$\sigma(c_p)$: $\sigma(x) \in p \in P$\}, where \{\text{$c_p$: $p \in P$}\} is a set of new distinct constants, is consistent.
\end{itemize}

3. $Q \subseteq \{p \in P: p$ is subtrivial and not trivial\}.

4. $F$ is a function from $Q$ to \{$0, \ldots, n - 1$\}.

5. $f^*$ is a partial function from $P$ to $P$ such that \{\text{$p$: $p$ is trivial or subtrivial}\} $\subseteq \text{Dom}(f^*)$ and the following conditions are satisfied:

\begin{itemize}
  \item if for some $d \in D$, "$f(x) = d'$" $\in p$, then $f^*(p)$ is defined and equal to the trivial type containing the basic formula "$x = d'$";
  \item for $p \neq p'$, if $f^*(p) = p'$ and $p'$ is not trivial, then for some $k < n$ and $d \in D$, "$x = f^k(d')" \in p$ and "$x = f^{k+1}(d')" \in p'$.
\end{itemize}

Let $\mathcal{A} = (A, a_1, \ldots, a_m, f)$ be a structure of the form (4) and let $c = (D, d_1, \ldots, d_m, P, Q, F, f^*)$ be a code. As follows from the definition of code, for each $d \in D$ there is exactly one 1-type $p \in P$ such that $p$ contains a formula "$x = d'$". We will denote this 1-type by $p_d$. Let $\delta$ be the function defined for $d \in D$ as follows: $\delta(d) = p_d$. We say that $\mathcal{A}$ admits $c$ if there exists a function $\epsilon$ from $P$ to
A such that the following conditions are satisfied, where $\varepsilon$ and $\delta$ denote also the induced translations of a basic 1-type:

1. $\{p : p$ is a basic 1-type on $\varepsilon \delta(D)$ realized in $A\} = \varepsilon \delta(P)$.
2. Each $p \in \varepsilon \delta(Q)$ is realized exactly $F(\delta^{-1}(\varepsilon^{-1}(p)))$ times in $A$. Each subtrivial but not trivial 1-type on $\varepsilon \delta(D)$ belonging to $\varepsilon \delta(P - Q)$ is realized at least $n$ times in $A$.
3. $ef^{*} \varepsilon^{-1} \subseteq f \cap (\varepsilon(P))^{2}$.
4. For each $p \in P$, $\varepsilon(p)$ realizes $\varepsilon \delta(p)$ in $A$.

**Lemma 9.** For every structure $A = (A, a_1, \ldots, a_m, f)$ there is a function $g$ and a code $c$ such that if $B = (A, a_1, \ldots, a_m, g)$, then $\text{TYP}(A) = \text{TYP}(B)$ and $B$ admits $c$.

**Proof.** In the case $\text{card}(A) \leq M(n, m)$ we take $f = g$, $D = A$ and $f_{E} = f$. The other elements of the code are determined in a natural way.

Assume now $\text{card}(A) > M(n, m)$. We consider all basic 1-types on the set $\{a_1, \ldots, a_m\}$ realized in $A$. For each such 1-type we choose as many elements realizing it as possible, up to maximum $n$. Moreover, with each chosen element we choose also all ‘associated’ elements. For example, if the chosen element $a$ realizes a 1-type $p$ such that $\theta \in p$, then we also choose elements $b_2, \ldots, b_n$ such that $t^{n}(a, b_2, \ldots, b_n) = t$. The procedure of choosing goes step by step. First we choose elements realizing trivial and subtrivial 1-types. Then we choose elements realizing cycles. Finally, we choose elements which realize other 1-types. Let $D_0$ denote the set of elements chosen in such a way. Now for $l \leq n$ we consider sets $B_l = \{a \in D_0 : f^l(a) \in D_0\}$ and $C_l = \{a \in D_0 : f(a) \notin D_0$ and for some $k < l, f^k(a) = f^l(a)\}$ and for each $a \in B_l \cup C_l$ we extend $D_0$ by adding elements $f(a), \ldots, f^{l-1}(a)$. We repeat this as many times as possible. Let $D$ be the set which arises in this way. The function $g$ on the set $D \cap f^{-1}(D)$ will coincide with $f$. We have to define the function $g$ on the other elements. First assume that $D$ is closed under the function $f$. We will consider some cases.

**Case 1:** there is a simple 1-type realized in $A$. If $A$ is finite, we define $g$ on $A - D$ putting $g(d_i) = d_{i+1}$ for $i = 1, \ldots, s - 1$ and $f(d_s) = d_1$ where $d_1, \ldots, d_s$ is a one-one sequence of all elements of $A - D$. If $A$ is infinite, then we define $g$ in such a way that the graph of this function on $A - D$ is a disjoint union of infinite chains.

**Case 2:** the first case does not hold and some 1-type being a cycle is realized in $A - D$. In this case we define $g$ on $A - D$ in such a way that the graph of $g$ will be a union of such cycles. This can only not be possible in the case that $A$ is finite. But in that case some element of $A - D$ must realize a ramifying 1-type being a cycle. Thus using less than $n$ times the graph of the following form
and adding it to the set $D$ we can define $g$ on the remaining elements as needed.

Case 3: if the above cases do not hold, then there is $a \in A$ such that $f^{-1}(a)$ has at least $n$ elements (here we use $\text{card}(A) \geq M(n, m) > (n - 1)^n(n + m)$). Thus, such an element belongs to $D$; let it be $b$. Then we define $g$ by putting $g(x) = b$ for all $x \in A - D$.

Assume now that $f(D)$ is not contained in $D$. Then some simple 1-type is realized in $A$. If $\mathfrak{A}$ is infinite, then we define $g$ in such a way that its graph on $A - D$ is a disjoint union of infinite chains. Similarly, for each $a \in D$ for which $g$ is not defined we extend the definition $g$ in such a way that the set $\{g'(a): l > 0\}$ is infinite and disjoint with $D$. Let us consider the case $\mathfrak{A}$ is finite. For each $a \in D$ such that $f(a) \notin D$ we have two possibilities:

- for some $k > n$, $f^k(a) \in D$ and for all $l < k$, $f^l(a) \notin D$;
- for some $l > n$ and $k$, $f^k(a) = f^l(a)$ and for all $l$, $f^l(a) \notin D$.

In the first case we take $b_1, \ldots, b_n \in A - D$ and put $g(a) = b_1$, $g(b_1) = b_{i+1}$ and $g(b_i) = f^k(a)$ and extend the set $D$ adding elements $b_1, \ldots, b_n$. In the second case we take $b_1, \ldots, b_n \in A - D$ and put $g(a) = b_1$, $g(b_1) = b_{i+1}$ and $g(b_i) = b_s$ where $s = \min(n, k)$ and $r = s + \min(n + 1, l - k)$. Similarly, we extend the set $D$ by adding elements $b_1, \ldots, b_r$. In this way $g$ is defined on $D$ (now extended) such that $g(D) \subseteq D$. On the set $A - D$ the function $g$ is defined in such a way that its graph forms a large circle. This finishes the definition of $g$ and at the same time of $\mathfrak{A}$.

Now we define a code $c = (D, d_1, \ldots, d_m, P_1, P_2, E, F, f*)$. The set $D$ was determined in the construction of $\mathfrak{A}$. An easy calculation shows that $\text{card}(D) \leq M(n, m)$. For $i = 1, \ldots, m$, $d_i = a_i$. $P$ is the set of all 1-types on $D$ realized in $\mathfrak{B}$. By our construction condition 2 in the definition of code is satisfied. $Q$ is the set of all subtrivial but not trivial 1-types on $D$ which are realized in $\mathfrak{B}$ not more than $n - 1$ times. $F$ is a function saying how many times a given 1-type from $Q$ is realized. Let $\delta$ be the function defined before the definition of admittance. Thus $f^*$ is such that $f^* \cap (\delta(D))^2 = \delta(g \cap D^2)$. On the other elements we define $f^*$ where possible according to condition 5 in the definition of code. Finally by an easy argument we show that $\mathfrak{B}$ admits a code. \[ \Box \]

Let a code $c = (D, d_1, \ldots, d_m, P, Q, F, f^*)$ be given. Let $k$ be the number of exciting basic 1-types in $P$. Let $(p_i: t \leq s)$ be the set of all basic 1-types which are neither trivial nor semitrivial and do not belong to $Q$. Finally let $Q = \{q_i: i \leq r\}$. For such a code $c$ we define the following sentence $\phi_c$:

$$\exists \lambda_0 \cdots \lambda_s \left( \phi^0_c \land \phi^s_c \right)$$

where $\phi^0_c$ is the formula

$$\left[ \sum_{i=0}^s \lambda_i + \sum_{i=0}^s F(q_i) + \text{card}(D) + \omega \cdot k = \text{card}(A) \right] \land \sum_{i=0}^s \lambda_i > n$$
and $\phi_i^j$ is the formula
\[
\bigwedge_{j=0}^n \left( \sum_{i=0}^{n} \left( \lambda_i \colon \text{$p_i$ is a $j$-cycle} \right) \right) \text{ is divisible by $j$ or infinite}.
\]

**Lemma 10.** If $\mathfrak{A}$ admits $c$, then $\mathfrak{A} \models \phi_c$.

**Lemma 11.** If $\mathfrak{A}^{-} = (A, a_1, \ldots, a_m)$ is such that $\mathfrak{A}^{-} \models \phi_c$, then there is a function $f$ such that $\mathfrak{A} = (\mathfrak{A}^{-}, f)$ admits $c$.

**Proof.** Let $\lambda_0, \ldots, \lambda_s$ be cardinal numbers witnessing that $\phi_c$ is true in $\mathfrak{A}^{-}$. By $\phi_0^0$ we can assume that
\[
A = \bigcup_{i \leq r} A_i \cup B \cup \bigcup_{i \leq s} C_i \cup D
\]
where all sets $A_0, \ldots, A_r, B, C_0, \ldots, C_s, D$ are pairwise disjoint, and such that $\text{card}(a_i) = F(q_i)$, $\text{card}(C_i) = \lambda_i$ and $\text{card}(B) = \omega \cdot k$. First we define the function $f$ on the set $D$ according to trivial types. If $k \neq 0$, then we use the set $B$ for defining $f$ in such a way that $B = \{f^i(d) : p_d \text{ is exciting and } l > 0\}$ and for each exciting $p_d$ the set $\{f^i(d) : l \in \omega\}$ is infinite. For $a \in A_i$ we put $f(a) = d$ where $d \in D$ is such that $\text{"'}f(x) = d\text{"'} \in q_i$. In the same way we define $f$ on $A_i$ if $p_i$ is subtrivial. If $p_i$ is an $r$-cycle for some $r \leq n$, then we define $f$ on $A_i$ in such a way that for all $a \in A_i$, $\{f^i(a) : l > 0\}$ has exactly $r$ elements. By $\phi_1^1$ this is possible. Finally, if $p_i$ is a simple 1-type and $\lambda_i$ is infinite, then we define $f$ such that its graph is a disjoint union of infinite chains. In the case $\lambda_i$ is finite we define $f$ such that for arbitrary $a \in A_i$, $\{f^i(a) : l \leq \lambda_i\} = A_i$. This finishes the definition of $f$.

To show that $\mathfrak{A}$ admits $c$ we define $\varepsilon$ in a natural way (i.e., $\varepsilon(p_d) = d$, $\varepsilon(p) \in A_i$, $\varepsilon(q_i) \in C_i$, and for semitrivial but not trivial $p$, $\varepsilon(p) = f^i(d)$ where $l$ and $d$ are such that $\text{"'}x = f^i(d)\text{"'} \in p$).

**Lemma 12.** If $\mathfrak{A}$ admits the code $c$, then $\text{TYP}^\mathfrak{A}(c)$ can be effectively determined from $c$.

**Proof.** A type $t$ determines some directed graph $G_t$ having at most $2n$ elements. On the other hand a code determines also some finite graph $G_c$ — the graph of the function $f$ restricted to $D$. To decide if a type $t$ belongs to $\text{TYP}^\mathfrak{A}(c)$ we must verify if the graph $G_t$ is a subgraph of the graph $G_c$. Such a verification is effective.

By the last lemma it makes sense to write $\text{TYP}(c)$.

A basic sentence is a sentence of the form
\[
\text{card}(A) \geq \omega \land \phi \quad \text{or} \quad \text{card}(A) < \omega \land \psi(\text{card}(A))
\]
where $\phi$ is a sentence and $\psi(x)$ is a formula of the language of arithmetic with addition.
Lemma 13. For every code $c$ there is a disjunction of basic sentences which is equivalent to $\phi_c$.

Proof. The sentence $\phi_c$ is equivalent to the following sentence

$$[\text{card}(A) < \omega \land \exists \lambda_0 \cdots \lambda_s \phi'(\text{card}(A))]
\lor \left[\text{card}(A) \geq \omega \land \left(\bigvee_{(i_0, \ldots, i_s) \in \mathbb{N}^{s+1}} \exists \lambda_{i_0} \cdots \lambda_i \phi_{K,R}^{\lambda_i}\right)\right]$$

where the disjunction $\bigvee_{R} \exists \lambda_{i_0} \cdots \lambda_i \phi_{K,R}^{\lambda_i}$ is taken over all possible orderings $R$ of the set $(s + 1) - \{i_0, \ldots, i_s\}$. $\phi'(\text{card}(A))$ is the formula $\phi^0 \land \phi^1$ in which $\lambda_0, \ldots, \lambda_i$ are free variables, and $\phi_{K,R}^{\lambda_i}$ arises from $\phi^0 \land \phi^1$ by interpreting $\lambda_{i_0}, \ldots, \lambda_i$ as number variables and the others as fixed cardinal numbers ordered with respect to their value by the relation $R$. Here we only use that the sum of the infinite cardinal numbers is equal to the largest one. This gives us the required disjunction of the basic sentences. $\square$

Theorem 7. The theory of equality in $L(F_n)$ is decidable.

Proof. We will show that each sentence is effectively equivalent to some boolean combination of basic sentences. We construct it by elimination of quantifiers. It is enough to show that if $\phi$ is a boolean combination of atomic formulas and basic sentences, then the formula $F_n x_1 \cdots x_n y_1 \cdots y_n \phi$ is equivalent to some boolean combination of atomic formulas and basic sentences. We can assume that $\phi$ has the form $\bigvee_i \bigwedge_j \phi_{ij}$ where each $\phi_{ij}$ is an atomic formula or negation of an atomic formula or a basic sentence. In a standard way (see Krynicki–Lachlan [9, p. 193]) we can ‘eliminate’ basic sentences and atomic formulas not containing the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$. In effect, we can assume that the free variables of $\phi$ are $x_1, \ldots, x_n, y_1, \ldots, y_n, z_1, \ldots, z_m$. Moreover, in each part of the disjunction there is a conjunction $\bigwedge_i z_i \neq z_j$. Now we proceed as in the proof of Fact 2.6 in Krynicki–Lachlan [9]. Each type $t$ of $n$ elements is characterized by a quantifier-free formula $\phi_t$. We can assume that for each $t$, $\phi_t$ implies $\bigwedge_i z_i \neq z_j$. We denote $\text{TYP}^n(\phi) = \{t: \phi_t$ implies $\phi\}$. Let $C = \{c: c$ is a code and $\text{TYP}^n(c) \subseteq \text{TYP}^n(\phi)\}$. By Lemmas 10 and 11 for arbitrary $A$ and pairwise different elements $a_1, \ldots, a_m$ of $A$ we have:

$$(A, a_1, \ldots, a_m) \models F_n x_1 \cdots x_n y_1 \cdots y_n \phi[a_1, \ldots, a_m] \iff (A, a_1, \ldots, a_m) \models \bigvee_{c \in C} \phi_c.$$

By Lemma 13, $\bigvee_{c \in C} \phi_c$ is equivalent to a disjunction of basic sentences. This finishes the proof of Theorem 7. $\square$

Corollary 1. The theory of equality in $L(F_m)$ is decidable.

The last two results allow us to give a partial answer to the question stated in Krynicki–Väänänen [10]. Namely, by Corollary 1 and Theorem 1 we have:
Corollary 2. (a) $L(F_\omega)$ is not recursively semantically equivalent to $L(H_\omega)$.
(b) $L(F_\omega)$ is not recursively semantically equivalent to $L(F^*_\omega)$.
(c) $L(F_\omega)$ is not recursively semantically equivalent to $L^*$.

Finally we can deduce the following result implicitly contained in Krynicki–Vaäänänen [10].

Corollary 3. The Hanf and Skolem numbers for the theory of equality in $L(F_n)$, as well as in $L(F_\omega)$, are $\omega$.

From the proof of the main theorem we can also deduce the following corollary.

Corollary 4. Each sentence of the theory of equality in $L(F_\omega)$ is equivalent to a sentence of the form
$$\text{card}(A) > \omega \lor (\text{card}(A) < \omega \land \psi(\text{card}(A)))$$
where $\psi(x)$ is a disjunction of the conjunctions of the following formulas: $x < i$, $j < x$, $x = k$, $x = m \ (\text{mod } n)$. Moreover, this sentence can be found in an effective way.

Let $D_n$ be a divisibility quantifier whose semantical meaning is defined as follows:
$$D_n x \phi \iff \text{card}\{x: \phi(x)\} \text{ is divisible by } n \text{ or infinite.}$$
The quantifier $D_n$ is definable by the quantifier $F_n$. Let $L(D_\omega)$ denote the logic with divisibility quantifiers $D_n$ for all $n \in \omega$. An analysis of the proof of Theorem 7 and some simple number-theoretic facts allows us to prove (see also Mostowski [15]) the following corollary.

Corollary 5. (a) The theory of equality in $L(F_\omega)$ and the theory of equality in $L(D_\omega)$ are recursively equivalent (i.e., for every formula of the theory of equality in $L(F_\omega)$ there can effectively be found a semantically equivalent formula of the theory of equality in $L(D_\omega)$).
(b) The theory of equality in $L(F_n)$ is recursively semantically equivalent to the theory of equality in $L(D_k)_{k<n}$.

3. Some other theories

3.1. General lemma

In this section we extend some results of Krynicki–Lachlan [9] concerning decidability and undecidability of theories in $L(H)$. In that paper it was proved that the theory of unary relations is decidable in $L(H)$. On the other hand the
theory of successor functions is undecidable in $L(F_2)$. The proof of this fact is based on the observation that in $(\omega, s)$, where $s$ is the usual successor function, we can define addition and multiplication. This allows us also to prove the following lemma, suggested by A.H. Lachlan:

**Lemma 14.** Let us assume that there is a formula $\phi(x, y)$ and a model $\mathcal{A}$ for $T$ such that for any $n \in \omega$ there is $b \in |\mathcal{A}|$ such that there is a finite number, but more than $n$, of elements $a \in |\mathcal{A}|$ for which $\phi(a, b)$ holds in $\mathcal{A}$. Then $T$ is undecidable in $L(F_2)$.

**Proof.** Let $\phi$ be a formula satisfying the assumptions of the lemma. We define:

- $U(x)$ iff $\{z: \phi(z, x)\}$ is finite,
- $R(x, y)$ iff $\text{card}\{z: \phi(z, x)\} = \text{card}\{z: \phi(z, y)\}$,
- $S(x, y)$ iff $\text{card}\{z: \phi(z, x)\} < \text{card}\{z: \phi(z, y)\}$.

Notice that $U(x)$ is expressible in $L(F_2)$ (this can be done by means of the Ehrenfeucht sentence restricted to a proper formula). $R(x, y)$ and $S(x, y)$ are also expressible in $L(F_2)$ (see, e.g., Krynicki–Lachlan [9]). We denote by $\sigma$ the sentence $\exists x \, U(x) \wedge \forall x \, \exists y \, (U(x) \rightarrow U(y) \wedge S(x, y))$, and $T' = T \cup \{\sigma\}$. By our assumption $T'$ is consistent. Moreover, if $\mathcal{A}$ is a model for $T'$, then we have $(U^\mathcal{A}/R^\mathcal{A}, S^\mathcal{A}/R^\mathcal{A}) = (\omega, <)$. Now, as in Krynicki–Lachlan [9, Lemma 3.2], we can show that addition and multiplication are categorically definable. Thus $T'$ and consequently $T$ is undecidable in $L(F_2)$. □

**Corollary 6.** The following theories are undecidable in $L(F_2)$:
- the theory of one unary function,
- the theory of boolean algebras,
- the theory of ordered groups,
- the theory of archimedian order groups.

**Proof.** We apply Lemma 14 using the formula $y = f(x)$ in the first case, “$x$ is an atom and $x \leq y$” in the second case, and $0 < x \wedge x < y$ in the last two cases. □

Some results from the last corollary can be obtained using the undecidability of the considered theories in the language with the Härting quantifier (see Herre et al. [7]).

### 3.2. Group theory

Now we consider the theory of groups. First we define the following formula:

$$\phi(x, y) = \neg F_2uvzt [(u = e \rightarrow z = e) \wedge (u = v + y \rightarrow z = t + y) \wedge (u = x \rightarrow z \neq x)],$$

where $e$ denotes the neutral element of a group.
Lemma 15. For every group $\mathcal{A} = (G, +, 0)$ and $a, b \in G$ we have $\mathcal{A} \models \phi(a, b)$ if and only if there is $n \in \mathbb{Z}$ such that $a = nb$.

Proof. ($\Rightarrow$) Let us assume that $\mathcal{A} \models \neg \phi(a, b)$. Then there is a function $f : G \to G$ such that $f(0) = 0$, and for every $g \in G$, $f(g + b) = f(g) + b$. Thus for every $n \in \mathbb{Z}$, $f(nb) = nb$. But $f(a) \neq a$ so there is no $n \in \mathbb{Z}$ such that $a = nb$.

($\Leftarrow$) Let us assume that $a \neq nb$ for any $n \in \mathbb{Z}$. We define $f : G \to G$ step by step in the following way. First we put $f(nb) = nb$ for all $n \in \mathbb{Z}$. Now let $f$ be defined on $H$ being the proper subset of $G$ such that for all $x \in H$ and $n \in \mathbb{Z}$, $x + nb \in H$. Now let $x \in G - H$. We extend $f$ by putting $f(x + nb) = nb$ for each $n \in \mathbb{Z}$. This shows that $\mathcal{A} \models \neg \phi(a, b)$. □

Corollary 7. The following notions are expressible in $L(F_2)$:

- $G$ is a cyclic group.
- $G$ is a torsion-free group.
- $G$ has an element of infinite order.
- An element $x$ is of finite order.
- An element $x$ is a generator of a group $G$.

Proof. By Lemma 15 each of these sentences can be expressed using the formula $\phi$. □

Theorem 8. Every extension of the theory of groups having a model with an element of infinite order is undecidable in $L(F_2)$.

Proof. Let $T$ be a theory satisfying the assumptions of the theorem. Let $T' = T \cup \{\exists y \ Q_{\omega}x (x, y)\}$, where $Q_{\omega}$ is the quantifier ‘there exist infinitely many’ and $\phi$ is the formula defined before Lemma 15. Thus $T'$ is consistent. Let $c$ be a new constant and let $T_i = T' \cup \{Q_{\omega}x \phi(x, c)\}$. We define:

$$U(x) = \phi(x, c), \quad S(x, y) = y = x + c,$$

$$U^+(x) \equiv U(x) \land F_2 \forall u \forall v ((u = z \Rightarrow v = t) \land (u = 0 \Rightarrow v = 0) \land (u \neq 0 \land U(u) \Rightarrow (v = u + c \lor v + c = u)) \land (v = z \Rightarrow (u \neq t \lor u = 0)) \land (u = x \Rightarrow v = u + c)).$$

By our assumptions $U(x)$ and $S(x, y)$ define a structure isomorphic to the integers with successor function. $U^+(x)$ is satisfied if $U(x)$ and there is a one-one function $f$ such that $f(0) = 0$, for any $y \neq 0$ from $U$ either $f(y) = y + c$ or $f(y) = y - c$, for no $y \neq 0$ $f(y) = y$, and moreover $f(x) = x + c$. Therefore $U^+$ defines the positive part of $U$. Hence in any model for $T_i$, $U^+(x)$ and $S(x, y)$ define a system isomorphic to $(\omega, s)$. Thus $T_i$ and consequently $T$ are undecidable. □

The last theorem gives us many examples of undecidable theories in $L(F_2)$: the theory of abelian groups, the theory of cyclic groups, the theory of free torsion groups etc.
Let us turn to the case of the theory of groups with all elements of finite order. Here we can prove the following:

**Theorem 9.** Every theory of groups having a model with elements of arbitrary large finite order is undecidable in $L(\mathbb{F}_2)$.

**Proof.** Using the formula $\phi(x, y)$ we can define the relation "the order of an element . . . is less than the order of an element . . ." between elements of finite order. This allows us to interpret $(\omega, <)$ in a model of our theory. \(\square\)

In Baudisch [1] it was proved that the theory of $p$-adic groups in the language with the Hártig quantifier is decidable. In spite of this we obtain the following corollary.

**Corollary 8.** The theory of $p$-adic groups is undecidable in $L(\mathbb{F}_2)$.

3.3. *The theory of fields*

The same method as in the case of the theory of groups can be applied to many theories of fields. Let $\sigma_1$ be the formula $\forall x \phi(x, 1)$ and $\sigma_2$ the formula $\forall x \exists y (\phi(y, 1) \& x < y)$ where $\phi$ is the formula defined before Lemma 15.

**Lemma 16.** (a) For an arbitrary field $K$ we have $K \models \sigma_1$ iff $K$ is of characteristic 0.

(b) For arbitrary ordered field $K$ we have $K \models \sigma_2$ iff $K$ is an archimedian ordered field.

**Theorem 10.** Each extension of the theory of fields having as a model some field of characteristic 0 is undecidable in $L(\mathbb{F}_2)$.

**Proof.** The idea is the same as in the proof of Theorem 8. The difference is that we use the constant 1 instead of the parameter $c$. \(\square\)

Thus again we obtain several examples of undecidable theories, for instance: the theory of fields, the theory of real closed fields, the theory of algebraic closed fields etc.

3.4. *The theory of one equivalence relation*

Now let us consider the theory of one equivalence relation in $L(\mathbb{F}_2)$. We can prove the following:

**Theorem 11.** If $T$ is an extension of the theory of one equivalence relation having a model in which there are infinitely many equivalence classes of distinct power, then $T$ is undecidable in $L(\mathbb{F}_2)$. 

Proof. Using the definability of the Hartig quantifier in $L(F_2)$, we can in a standard way interpret the theory of $(\omega, <)$ in $T$. This gives us the undecidability of $T$. □

As a complement to the last theorem we give:

Remark. A theory of one equivalence relation having at most $n$ equivalence classes is decidable in $L(H)$ and hence in $L(F_2)$.

Proof. It is easy to see that such a theory is interpretable in the theory of unary relations. □

3.5. The theory of infinitely many monadic predicates

Until now we have considered Henkin quantifiers of the form $H^1_n$ only. The question whether the restriction to such quantifiers is essential remains open. However, we know by Theorem 1 that every quantifier $H^2_m$ can be expressed using identity in infinite domains by means of a quantifier $H^2_m$ (for suitable $m$). By the monadic logic of a logic $L$ we mean the theory of infinitely many unary relations with identity in the logic $L$. In [9] it was shown that the monadic logic of $L(H)$ is decidable. Here we shall prove that it is not so in the case of $L(H_2^2)$. Actually we will prove a stronger theorem for $F_2$.

Theorem 12. The monadic logic of $L(F_2^2)$ is undecidable.

Proof. It was shown by Matijasevič [11] that: “there is no algorithm for determining whether an arbitrary diophantine equation has a solution”. From this follows the undecidability of the set of true existential statements about natural numbers built up using constant 0, operations $s, +, \cdot,$ and identity symbols $=.$ Reducing the truth problem for such statements to the tautology problem for the monadic logic of $L(F_2^2)$, we can show that the monadic logic of $L(F_2^2)$ is undecidable.

Let $\phi$ be the statement $\exists x_1 \cdots \exists x_n \alpha(x_1, \ldots, x_n)$, where $\alpha(x_1, \ldots, x_n)$ is a quantifier-free formula built up using constant 0, operations $s, +, \cdot,$ and identity symbol $=,$ with no other variables than $x_1, \ldots, x_n$. First we translate the formula $\alpha(x_1, \ldots, x_n)$ into a statement in the monadic language with $F_2^2$. The idea of the translation is as follows: a numerical variable $x_i$ will be represented by a predicate $P_i$, a value of $x_i$ will be represented by a number of elements satisfying $P_i$; all numerical relations between $x_1, \ldots, x_n$ will be represented by corresponding numerical relations between powers of $P_1, \ldots, P_n$; at the end we join the obtained statement into conjunction with the Ehrenfeucht statement saying that the universe is finite. In this way we obtain a sentence $\phi^*$ in the monadic language with $F_2^2$, such that $\phi$ is true if and only if $\phi^*$ has a model (which has to be finite).
It remains to define the translation. We assume that \( \alpha(x_1, \ldots, x_n) \) is a boolean combination of atomic formulas: \( x_i = x_j, x_i = 0, x_i = s(x_j), x_i = x_j + x_k, x_i = x_j \cdot x_k \). We can obtain a formula satisfying this assumption in an effective way from \( \phi \) increasing eventually the number of existential quantifiers. Then we define the translation for atomic formulas. We shall give a translation into second-order logic, keeping in mind that all second-order quantifiers can be eliminated in favour of \( F_{\mathbb{Z}} \). Let \( \mathbf{L}(\phi(x, y), \psi(x, z)) \) be the abbreviation of the following formula

\[
\exists y \forall x' \forall x' \left[ (\phi(x, y) \leftrightarrow \psi(f(x), z)) \land (\psi(x, z) \leftrightarrow \phi(f(x), y)) \right] \land ((f(x) = f(x')) \Rightarrow x = x').
\]

\( x_i = x_j \) is replaced by

\( \mathbf{L}(P_i(x), P_j(x)) \).

\( x_i = 0 \) is replaced by

\( \forall x \neg P_i(x) \).

\( x_i = s(x_j) \) is replaced by

\( \exists z P'_i(z) \land \mathbf{L}(P_i(x) \land x \neq z, P_j(x)) \).

\( x_i = x_j + x_k \) is replaced by

\( \mathbf{L}(P_i(x), P_j(x) \lor P_k(x)) \).

\( x_i = x_j \cdot x_k \) is replaced by

\[
\exists y \forall x' \forall y' \left[ (P_i(x) \land P_j(y) \land P_k(x') \land P_k(y') \land f(x, x') = f(y, y')) \Rightarrow (x = y \land x' = y') \land (P_i(f(x, x')) \leftrightarrow (P'(x) \land P_k(x')) \land P_j(z) \land z \neq f(x, x')) \right].
\]

For the adequacy of this translation (particularly in the case of addition) we have to guarantee the disjointness of \( P_i \) and \( P_j \) for \( i \neq j \). Then we define the effect of the translation as the conjunction of the formula obtained by the above replacements and the formula being the conjunction of formulas \( \forall x \neg (P_i(x) \land P_j(x)) \), for \( i, j \) such that \( i \neq j \) and \( P_i, P_j \) occurring in the translated formula.

**Theorem 13.** The theory of infinitely many unary relations in \( L(H_u) \) is undecidable.

**Proof.** By the proof of the previous theorem it suffices to define a translation for \( x_i = x_j \cdot x_k \), that is a formula saying that the number of \( x \) satisfying \( P_i(x) \) is equal to the number of \( x \) satisfying \( P_j(x) \) multiplied by the number of \( x \) satisfying \( P_k(x) \). We use the property that \( \text{card}(X) \leq \text{card}(Y \times Z) \) if and only if there are projections \( f : X \rightarrow Y, g : X \rightarrow Z \), such that for \( x, y \in X : x = y \) if and only if
Decidability problems in languages with Henkin quantifiers

\[ f(x) = f(y) \land g(x) = g(y). \]
If \( X(Y, Z) \) is the set of elements satisfying \( \phi(x)(\psi(x), \chi(x) \) respectively), then this property can be expressed by the following formula, denoted by \( \Phi(\phi(x), \psi(x), \chi(x)) \):

\[
\exists f \exists f' \exists g \exists g' \forall x \forall x' \forall y \forall y'
\]

\[
((\phi(x) \land \psi(x') \rightarrow (x = x' \iff f(x) = f'(x')) \land \psi(f(x)))
\land (\phi(y) \land \phi(y') \rightarrow (y = y' \iff g(y) = g'(y')) \land \chi(g(y)))
\land (\phi(x) \land \phi(y) \rightarrow (x = y \iff (f(x) = f'(y) \land g(x) = g'(y)))).
\]

The above formula can easily be formulated in \( L(H_a) \). The translation of the formula \( x_1 = x_1 \cdot x_k \) will be the following formula \( \Phi(P_i(x), P_j(x), P_k(x)) \land \forall y (\Phi(P_i(x) \lor x = y, P_j(y), P_k(x)) \Rightarrow P_i(y)). \)

4. Final remarks

In this part we would like to discuss some unsolved problems and list some open questions.

1. We have proved that the theory of identity in \( L(H_a) \) is undecidable. It implies that this theory in a language with all Henkin quantifiers \( L^* \) is also undecidable, but we do not know if the language with all Henkin quantifiers is semantically more powerful than that of \( L(H_a) \), that is if \( L^* \neq L(H_a) \). It seems that \( L(H_a) \) is essentially weaker than \( I^* \). However there are Henkin quantifiers which seem to be weaker than those of the form \( H_n \), they are

\[
\forall \cdots \forall \exists \cdots \exists \quad \forall \exists \cdots \exists
\]

and

More generally, if \( A \) is an infinite class of nonlinear Henkin quantifiers then: what is the relation between \( L(A) \), \( L(H_a) \), and \( L^* \)?

2. All undecidability results for the theory of identity stated in this paper, particularly for \( L(H_a) \), essentially use the fact that the considered class of quantifiers are infinite. This does not exclude the possibility that the theory of identity in \( L(H_a) \) is decidable for all \( n \) (for \( n = 2 \) this was proved in [9]). If for some \( n \) the theory of identity in \( L(H_a) \) is undecidable, then there is the question of the minimal \( n \) with this property. A relevant result about unprovability of decidability of the theory of identity in \( L(Q) \), has been stated in [13], but it is not obvious how to conclude just undecidability from that.

3. Let \( \alpha \) be a signature with infinitely many unary predicates. We know that \( L_\alpha(H) \) is decidable, but \( L_\alpha(H_a) \) and \( L_\alpha(F_2^2) \) are undecidable. What about \( L_\alpha(H_3) \)? What is the situation if \( \alpha \) were finite? And finally, is \( L_\alpha(F_\omega) \) decidable?

4. For languages as powerful as the language with all Henkin quantifiers the borderline between axiomatizable (recursively enumerable) and not axiomatizable seems to be more interesting than that between decidable and undecidable.
In [14], it was stated that for every recursive set of Henkin quantifiers $A$ if the theory of identity in $L(A)$ is axiomatizable, then it is decidable. Does the same hold for every signature? For a signature $\sigma$ with at least one binary predicate or one unary function symbol $L_\sigma(H)$ is not recursively enumerable because full elementary arithmetic can be interpreted in $L_\sigma(H)$ (see [9]). We know that every nonlinear Henkin quantifier has to contain $H$, therefore the problem is reduced to the question: are there any decidable classes of Henkin quantifiers $A$ such that the theory of unary relations in $L(A)$ is recursively enumerable but undecidable?

References