On the Variable Hierarchy of First-Order Spectra

ERYK KOPCZYŃSKI, University of Warsaw
TONY TAN, Hasselt University and Transnational University of Limburg

The *spectrum* of a first-order logic sentence is the set of natural numbers that are cardinalities of its finite models. In this article, we study the hierarchy of first-order spectra based on the number of variables. It has been conjectured that it collapses to three variables. We show the opposite: it forms an infinite hierarchy. However, despite the fact that more variables can express more spectra, we show that to establish whether the class of first-order spectra is closed under complement, it is sufficient to consider sentences using only three variables and binary relations.

Categories and Subject Descriptors: F.1.3 [**Complexity Measures and Classes**]: Complexity Hierarchies; F.4.1 [**Mathematical Logic**]: Finite Model Theory

General Terms: Theory

Additional Key Words and Phrases: First-order spectra, bounded number of variables, nondeterministic exponential time

ACM Reference Format:
Eryk Kopczyński and Tony Tan. 2015. On the variable hierarchy of first-order spectra. ACM Trans. Comput. Logic 16, 2, Article 17 (April 2015), 12 pages.
DOI: http://dx.doi.org/10.1145/2733376

1. INTRODUCTION

The spectrum of a first-order sentence Φ (with the equality predicate), denoted by $\textsc{Spec}(\Phi)$, is the set of natural numbers that are cardinalities of finite models of Φ. More formally, $\textsc{Spec}(\Phi) = \{n \mid \Phi \text{ has a model with universe of cardinality } n\}$. A set is a *spectrum* if it is the spectrum of a first-order sentence. We let $\textsc{Spec}$ denote the class of all spectra. Without the equality predicate, it is known that if a sentence has a model of cardinality n, then it also has a model of cardinality $n + 1$.

The notion of the spectrum was introduced by Scholz, where he also asked whether there exists a necessary and sufficient condition for a set to be a spectrum [Scholz 1952]. Since its publication, Scholz's question and many of its variants have been investigated by many researchers for the past 60 years. Arguably, one of the main open problems on spectra is the one asked by Asser, known as Asser's conjecture, whether the complement of a spectrum is also a spectrum [Asser 1955].

Although seemingly unrelated, it turns out that the notion of spectra has a tight connection with complexity theory. In fact, Asser's conjecture is shown to be equivalent

The first author is supported by the Polish National Science Centre grant DEC 2012/07/D/ST6/02435. The second author is supported by FWO Pegasus Marie Curie fellowship.
Authors' addresses: E. Kopczyński, University of Warsaw; email: erykk@mimuw.edu.pl; T. Tan, Hasselt University and Transnational University of Limburg; email: ptony.tan@gmail.com.

DOI: http://dx.doi.org/10.1145/2733376

to the problem NE versus CO-NE,[1] when Jones and Selman, as well as Fagin independently showed that a set of integers is a spectrum if and only if its binary representation is in NE [Jones and Selman 1974; Fagin 1973, 1974]. It also immediately implies that if Asser's conjecture is false, i.e., there is a spectrum whose complement is not a spectrum, then NP $\neq$ CO-NP, hence NP $\neq$ P.

In this article, we study the following hierarchy of spectra, which we call the variable hierarchy: for every integer $k \geq 1$, define

$$\textsc{Spec}_k \;=\; \{\textsc{Spec}(\Phi) \mid \Phi \text{ uses only up to } k \text{ variables}\}.$$

Obviously, we have $\textsc{Spec}_1 \subseteq \textsc{Spec}_2 \subseteq \cdots$. It was conjectured that the variable hierarchy collapses to three variables, because three variables are enough to describe the computation of a Turing machine. For more discussion on this conjecture, we refer the reader to a recent survey by Durand et al. [2012].

In this article, we show the opposite: the variable hierarchy has an infinite number of levels. In other words, for every $k \geq 3$, $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$ (Corollary 4.2). Here we should note that it is already known that $\textsc{Spec}_1 \subsetneq \textsc{Spec}_2 \subsetneq \textsc{Spec}_3$. More discussion is provided in the next section.

Our proof follows from the following observations:

—To describe a computation of a nondeterministic Turing machine with runtime $O(N^k)$—for a fixed integer $k \geq 1$—with a first-order sentence acting on a structure of cardinality N, $2k+1$ variables are sufficient.

—Conversely, for each first-order sentence Φ with k variables, checking whether a structure of cardinality N is a model of Φ can be done on a nondeterministic Turing machine in time $O(N^k(\log N)^2)$ [Grandjean 1984, 1985; Grandjean and Olive 2004].

Curiously, despite the infinity of the variable hierarchy, by standard padding argument, our proof implies that the class of first-order spectra is closed under complement if and only if the complement of every spectrum of three-variable sentence (using only binary relations) is also a spectrum (Corollary 3.5). This means that to settle Asser's conjecture, it is sufficient to consider only three-variable sentences using only binary relations.

This article is organized as follows. In Section 2, we discuss some related results. In Section 3, we present a rather loose hierarchy: for every integer $k \geq 3$, $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+3}$. Then in Section 4, we show that by more careful bookkeeping, we obtain a tighter hierarchy: for every integer $k \geq 3$, $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$. In Section 5, we briefly discuss how our results can be translated to the setting of generalized spectra. We conclude in Section 6.

2. RELATED WORKS

In this section, we briefly review the spectra problem and discuss some related results. We refer the reader to a recent survey by Durand et al. [2012] for a more comprehensive treatment on the spectra problem and its history. Fagin's paper [Fagin 1993] covers the relation between the spectra problem and finite model theory and its connection with descriptive complexity nicely.

First, we remark that our result $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$, for each integer $k \geq 3$ complements the previous known result that $\textsc{Spec}_1 \subsetneq \textsc{Spec}_2 \subsetneq \textsc{Spec}_3$ [Durand et al. 2012], which can be proved as follows. First, a model of first-order sentence with only one variable remains a model after cloning elements, and thus $\textsc{Spec}_1$ only includes the empty set

[1]NE is the class of languages accepted by a nondeterministic (and possibly multitape) Turing machine with runtime $O(2^{kn})$ for some constant $k > 0$.

and sets of form $\{n : n \geq k\}$. In another article, we show that the class of spectra of two-variable logic with counting quantifiers is exactly the class of semilinear sets and closed under complement [Kopczyński and Tan 2015]. Using the same methods, one can show that SPEC_2 is the class of finite and cofinite sets, thus separating SPEC_2 from SPEC_1. On the other hand, three variables are enough to simulate an arbitrary Turing machine, so it is not difficult to construct a set in SPEC_3 that is not even semilinear, say, for example, $\{n^2 \mid n$ is the length of an accepting run of a Turing machine $M\}$, hence separating SPEC_3 from SPEC_2.

Related to the variable hierarchy is the arity hierarchy. Let $\text{SPEC}(\text{arity } k)$ denote the spectra of first-order sentences using only relations of arity at most k. Fagin showed that if there exists k such that $\text{SPEC}(\text{arity } k) = \text{SPEC}(\text{arity } k+1)$, the arity hierarchy collapses to k [Fagin 1975].

Lynch [1982] showed that $\text{NTIME}[N^k] \subseteq \text{SPEC}(\text{arity } k)$, where $\text{NTIME}[N^k]$ denotes the class of sets of positive integers (written in unary form) accepted by nondeterministic multitape Turing machine in time $O(N^k)$, where N is the input integer. The converse is still open and seems difficult. A proof for $\text{SPEC}(\text{arity } k) \subseteq \text{NTIME}[N^k]$ seems to require that model checking for first-order sentences (of arity k) on structures with universe of cardinality N can be done in $\text{NTIME}[N^k]$. However, a result by Chen et al. states that checking whether a graph of N vertices contains a k-clique, which is of constant arity 2, cannot be done in time $O(N^{o(k)})$ unless the exponential time hypothesis fails [Chen et al. 2004, 2006; Impagliazzo and Paturi 1999].

Grandjean, Olive, and Pudlak provide another body of related works that established the variable hierarchy for spectra of sentences using relation and function symbols [Grandjean 1984, 1985, 1990; Grandjean and Olive 2004; Pudlák 1975]. Let F-SPEC_k denote the spectra of first-order sentences using up to k variables with vocabulary consisting of relation and function symbols, and let $\text{F-SPEC}(k\forall)$ denote the restriction of F-SPEC_k to sentences written in prenex normal form with universal quantifiers only and using only k variables. In his series of papers, Grandjean showed that $\text{NRAM}[N^k] = \text{F-SPEC}(k\forall)$, for each positive integer k, where $\text{NRAM}[N^k]$ denotes the class of sets of positive integers accepted by nondeterministic RAM in time $O(N^k)$ and N is the input integer [Grandjean 1984, 1985, 1990]. By Skolemisation, it is shown that $\text{F-SPEC}_k = \text{F-SPEC}(k\forall) = \text{NRAM}[N^k]$ for all $k \geq 1$ [Grandjean and Olive 2004, Theorem 3.1]. Combined with Cook's hierarchy of nondeterministic time [Cook 1973] and the known inclusions $\text{NTIME}[T(n)\log T(n)] \subseteq \text{NRAM}[T(n)] \subseteq T(n)\log^2 T(n)$, for each function $T(n) \geq n$ (see Grandjean [1985]), it implies $\text{F-SPEC}_k \subsetneq \text{F-SPEC}_{k+1}$, for all $k \geq 1$.

This does not imply our hierarchy here: $\text{SPEC}_k \subsetneq \text{SPEC}_{2k+2}$. Obviously, every function can be translated into a relation in first-order logic. However, such translation requires at least one new variable for each function. It is not clear whether there is a translation in which the number of new variables introduced depends only on the arity of the functions and not on the number of functions. At this point, we should also remark that $\text{F-SPEC}_k = \text{F-SPEC}(k\forall)$ can be much more expressive than SPEC_k. Take, for example, $k = 1$. The class SPEC_1 consists of only empty set and sets of the form $\{n, n+1, \ldots\}$, whereas the class $\text{F-SPEC}(1\forall)$ contains PRIMES, the set of prime numbers [Grandjean 1990].

3. AN EASIER HIERARCHY

For a positive integer N, we write $\mathsf{BINARY}(N)$ to denote its binary representation. Correspondingly, for a set $A \subseteq \mathbb{N}$, we write $\mathsf{BINARY}(A) \subseteq \{1, 0\}^*$ to denote the set of the binary representations of the numbers in A. To make a comparison between languages and sets of positive integers, for a function $T : \mathbb{N} \to \mathbb{N}$, we define $\text{NTIME}[T(n)]$ to be the class of sets of positive integers whose binary representations are accepted by a nondeterministic (possibly multitape) Turing machine with runtime $O(T(n))$. The class NE denotes $\bigcup_{k>0}$ NTIME$[2^{kn}]$.

Note that our definition implies that languages in NTIME$[T(n)]$ consist of strings that start with 1. This does not effect the generality of our results here. For every language L, we can define $L' = \{1\} \cdot L$, and any Turing machine that accepts L can be easily modified to one that accepts L' without any change in complexity.

In the following for a positive integer n, we let $[n] = \{0, 1, 2, \ldots, n-1\}$. The proof of the following proposition will set a framework that we will use again later in the proofs of Theorems 3.2 and 4.1.

PROPOSITION 3.1. NTIME$[2^n] \subseteq$ SPEC$_3$. *More precisely, for every set of positive integers A where* BINARY$(A) \in$ NTIME$[2^n]$*, there is a first-order sentence Φ using only three variables and binary relations such that* SPEC$(\Phi) = A$*.*

PROOF. The proof is via the standard encoding of an accepting run of a nondeterministic Turing machine with a square grid representing the space-time diagram. Let A be a set of positive integers, where $A \in$ NTIME$[2^n]$. Let M be a t-tape nondeterministic Turing machine accepting BINARY(A) in time $O(2^n)$ and space $O(2^n)$; or, equivalently, for every $N \in A$, M accepts BINARY(N) in time and space $O(N)$. By linear speed-up [Papadimitriou 1994, Theorem 2.2], we can assume that M accepts BINARY(N) in time and space $\leq N$. This assumes that N is big enough (greater than some N_0), and this is not a problem for spectra—numbers smaller than N_0 can always be considered on a case-by-case basis.

For $N \in A$, the accepting run of M on BINARY(N) can be described as a square-grid $[N] \times [N]$, where each point $(x, y) \in [N] \times [N]$ depicting cell x in time y is labeled according to the transitions of M. We will construct a first-order sentence Φ such that the models of Φ are precisely such grids encoded as first-order structures of the universe $[N]$ with binary relations representing the labels of points $(x, y) \in [N] \times [N]$, and therefore SPEC$(\Phi) = A$.

The sentence Φ will be a conjunction of *axioms* that confirm that various parts of the model work as expected. The proof will consist of two parts:

—Depicting the computation of M with just three variables. Essentially, in this part, we want to describe that the labels on the points $(x-1, y)$, (x, y), $(x+1, y)$ and the labels on its surrounding points $(x-1, y+1)$, $(x, y+1)$, and $(x+1, y+1)$ must "match" according to the transitions of M.
—Verifying that the input to M is the binary representation of the cardinality of the universe.

The details are as follows.

Depicting the computation of M with just three variables. We first declare a successor SUC and a total ordering $<$ on the universe using three variables; this allows us to identify the universe with $[N]$ and is done simply by adding the well-known total order and successor axioms to Φ. The predicates MIN(x) and MAX(x) state that x is the minimal and maximal element (0 and $N-1$), respectively.

For a formula $\phi(x, y)$ with two free variables x and y, we take the third variable z and define the operators $\Delta_h\phi(x, y)$, $\overline{\Delta}_h\phi(x, y)$, and $\Delta_v\phi(x, y)$, where h and v stand for horizontal and vertical, respectively, as follows:

$$\begin{aligned} \Delta_h\phi(x, y) &:= \forall z\ \mathsf{SUC}(x, z) \Rightarrow \phi(z, y) \\ \overline{\Delta}_h\phi(x, y) &:= \forall z\ \mathsf{SUC}(z, x) \Rightarrow \phi(z, y) \\ \Delta_v\phi(x, y) &:= \forall z\ \mathsf{SUC}(y, z) \Rightarrow \phi(x, z). \end{aligned}$$

It is straightforward to see that for every (x, y) when x is not the minimal and the maximal elements and y is not the maximal elements,

—$\underline{\Delta}_h\phi(x, y)$ holds if and only if $\phi(x + 1, y)$ holds;
—$\overline{\Delta}_h\phi(x, y)$ holds if and only if $\phi(x - 1, y)$ holds; and
—$\Delta_v\phi(x, y)$ holds if and only if $\phi(x, y + 1)$ holds.

Let the alphabet of M be Σ, and Q be the set of states of M. We will require the following relations to simulate the machine:

—$\mathsf{SYMBOL}^i_a(x, y)$, which holds if and only if the x-th cell of the i-th tape contains the symbol a at time y.
—$\mathsf{STATE}^i_q(x, y)$, which holds if and only if the head on the i-th tape at time y is over the x-th cell, and the state is q.

Now, to make sure that Φ depicts a computation of M correctly, we state the following. *On every "step"* $y = 0, \dots, N-1$*, if the heads are in states* $q_1, \dots, q_t$*, then for every cell* $x = 0, \dots, N-1$*, the labels on* $(x-1, y)$*,* (x, y)*,* $(x+1, y)$ *and the labels on* $(x-1, y+1)$*,* $(x, y+1)$*, and* $(x+1, y+1)$ *must "match" according to the transitions of* M.

Formally, it can be written as follows:

$$\bigwedge_{\bar{q}=(q_1,\dots,q_t)\in Q^t} \forall y \; \left(\left(\bigwedge_{1\le i\le t} \exists x \; \mathsf{STATE}^i_{q_i}(x, y) \right) \to \left(\bigwedge_{\phi} \forall x \; \phi(x, y) \;\to\; \psi_{\phi,\bar{q}}(x, y) \right) \right),$$

where the intuitive meaning of ϕ and $\psi_{\phi,\bar{q}}$ are as follows:

—The ϕ in the conjunction $\bigwedge_\phi$ runs through all possible labels of $(x-1, y)$, (x, y) and $(x+1, y)$, where each ϕ is of form

$$\overline{\Delta}_h \; \ell ab_1(x, y) \;\wedge\; \ell ab_2(x, y) \;\wedge\; \Delta_h \; \ell ab_3(x, y).$$

Intuitively, it means that $(x-1, y)$, (x, y), and $(x+1, y)$ are labeled with ℓab_1, ℓab_2, and ℓab_3, respectively, where each ℓab_1, ℓab_2, and ℓab_3 is a conjunction of the atomic relations STATE^i_q, and SYMBOL^i_a, as well as MIN and MAX, and their negations to indicate whether x or y is the minimal or maximal element.
—The formula $\psi_{\phi,\bar{q}}(x, y)$ is a disjunction of all possible labels on the points $(x-1, y+1)$, $(x, y+1)$, and $(x+1, y+1)$ according to the transitions of M, when the points $(x-1, y)$, (x, y), and $(x+1, y)$ satisfy ϕ and the states of the heads are $\bar{q} = (q_1, \dots, q_t)$. Formally, $\psi_{\phi,\bar{q}}(x, y)$ is of form

$$\psi_{\phi,\bar{q}}(x, y) \;:=\; \Delta_v\Big(\overline{\Delta}_h\psi'_{\phi,\bar{q}}(x, y) \;\wedge\; \psi''_{\phi,\bar{q}}(x, y) \;\wedge\; \Delta_h\psi'''_{\phi,\bar{q}}(x, y)\Big),$$

where $\psi'_{\phi,\bar{q}}, \psi''_{\phi,\bar{q}}, \psi'''_{\phi,\bar{q}}$ are all the disjunctions of all possible labels on $(x-1, y+1)$, $(x, y+1)$, and $(x+1, y+1)$, respectively, that are permitted by the transitions of M, when the points $(x-1, y)$, (x, y), and $(x+1, y)$ satisfy ϕ and the states of the heads are $\bar{q}$.

Of course, we also have to state that *for every step* $y = 0, \dots, N-1$*, there are only* t *heads*—that is, on every step $y = 0, \dots, N-1$, for every $i = 1, \dots, t$, there is exactly one cell x where (x, y) is labeled with STATE^i_q. This is straightforward.

Verifying the input to the Turing machine. The input will be provided in binary. Recall that the elements of universe correspond to the numbers from 0 to $N-1$. We will need the following axioms:

—The relation $\mathsf{DOUBLE}(x, y)$, which holds if and only if $x = 2y$. It is defined inductively by $x = y = 0$ and $(x - 2) = 2(y - 1)$:

$$\forall x \forall y \left(\begin{array}{c} \mathsf{DOUBLE}(x, y) \Leftrightarrow (\mathsf{MIN}(x) \wedge \mathsf{MIN}(y)) \vee \\ (\exists z\,(\mathsf{SUC}(z, x) \wedge \exists x\,(\mathsf{SUC}(x, z) \wedge \exists z(\mathsf{SUC}(z, y) \wedge \mathsf{DOUBLE}(x, z))))) \end{array} \right).$$

—The relation $\mathsf{HALF}(x, y)$, which holds if and only if $x = \lfloor y/2 \rfloor$—that is, $y = 2x$ or $y = 2x + 1$:

$$\forall x \forall y\,(\mathsf{HALF}(x, y) \Leftrightarrow \mathsf{DOUBLE}(y, x) \vee \exists z\,(\mathsf{DOUBLE}(z, x) \wedge \mathsf{SUC}(z, y))).$$

—The relation $\mathsf{DIV}(x, y)$, which holds if and only if $x = \lfloor (N-1)/2^y \rfloor$. It is defined inductively by $\lfloor (N-1)/2^0 \rfloor = N - 1$ and $\lfloor (N-1)/2^y \rfloor = \lfloor \lfloor (N-1)/2^{y-1} \rfloor /2 \rfloor$):

$$\forall x \forall y \left(\begin{array}{c} \mathsf{DIV}(x, y) \Leftrightarrow (\mathsf{MAX}(x) \wedge \mathsf{MIN}(y)) \vee \\ \exists z(\mathsf{SUC}(z, y) \wedge \exists y(\mathsf{DIV}(y, z) \wedge \mathsf{HALF}(x, y))) \end{array} \right).$$

—The relation $\mathsf{BIT}(y)$, which holds if and only if the bit b_y of the binary representation $b_{N-1} \cdots b_1 b_0$ of $N - 1$ is 1—that is, the integer $x = \lfloor (N-1)/2^y \rfloor$ is odd:

$$\forall y\,(\mathsf{BIT}(y) \Leftrightarrow \exists x(\mathsf{DIV}(x, y) \wedge \neg \exists z \mathsf{DOUBLE}(x, z))).$$

Finally, notice that because the relation BIT encodes the binary representation of $N-1$, the relation denoted by INPUT that encodes the input string (i.e., the binary representation of N) is defined by the following axiom:

$$\exists x \left(\neg \mathsf{BIT}(x) \;\wedge\; \mathsf{INPUT}(x) \;\wedge \left(\begin{array}{l} \forall y < x\,(\mathsf{BIT}(y) \wedge \neg \mathsf{INPUT}(y)) \wedge \\ \forall y > x\,(\mathsf{INPUT}(y) \Leftrightarrow \mathsf{BIT}(y)) \end{array} \right) \right).$$

This completes our proof of Proposition 3.1. □

Proposition 3.1 can be generalized to NTIME$[2^{kn}]$ as stated in the following theorem.

THEOREM 3.2. *For every integer $k \geq 1$,* NTIME$[2^{kn}] \subseteq$ SPEC$_{2k+1}$.

PROOF. The proof follows the same outline as the proof of Proposition 3.1. Let A be a set of positive integers such that $\mathsf{BINARY}(A) \in$ NTIME$[2^{kn}]$ and M be a t-tape nondeterministic Turing machine accepting $\mathsf{BINARY}(A)$ in time N^k and space N^k. So the space-time diagram is an $[N^k] \times [N^k]$ grid.

We identify numbers in $[N^k]$ with vectors $(p_k, p_{k-1}, \ldots, p_1) \in [N]^k$. The lexicographical successor relation $\mathsf{SUC}(p_k, \ldots, p_1, q_k, \ldots, q_1)$ can be defined as $1 + \sum_i p_i N^{i-1} = \sum_i q_i N^{i-1}$.

As in the proof of Proposition 3.1, the first-order sentence essentially states the following. *On every "step" $\bar{y} \in [N]^k$, if the heads are in states $q_1, \ldots, q_t$, then for every cell $\bar{x} \in [N]^k$, the labels on $(\bar{x}'', \bar{y})$, $(\bar{x}, \bar{y})$, and $(\bar{x}', \bar{y})$ and the labels on $(\bar{x}'', \bar{y}')$, $(\bar{x}, \bar{y}')$, and $(\bar{x}', \bar{y}')$ must "match" according to the transitions in M*, where $\bar{x}'$ and $\bar{y}'$ are the lexicographical successors of $\bar{x}$ and $\bar{y}$, respectively, and $\bar{x}''$ is the lexicographical predecessor of $\bar{x}$.

Accordingly, the relations SYMBOL_a^i and STATE_q^i are of arity $2k$. The only significant difference is the shift operators $\overline{\Delta}_h$, Δ_h, and Δ_v, which use only one extra variable, z, in their expansion. Let $\bar{x} = (x_k, \ldots, x_1)$ and $\bar{y} = (y_k, \ldots, y_1)$. The operator Δ_h is defined on any formula $\phi(\bar{x}, \bar{y})$ as follows:

$$\begin{array}{l} \Delta_h \phi(\bar{x}, \bar{y}) \;:=\; \displaystyle\bigvee_{i=2}^{k} \exists z \bigwedge_{j=1}^{i-1} \left(\begin{array}{c} \mathsf{MAX}(x_j) \;\wedge\; \mathsf{SUC}(x_i, z) \;\wedge \\ \exists x_1(\mathsf{MIN}(x_1) \wedge \phi(x_k, \ldots, x_{i+1}, z, x_1, \ldots, x_1, \bar{y})) \end{array} \right) \\ \qquad\qquad\qquad \vee\, (\exists z\; \mathsf{SUC}(x_1, z) \;\wedge\; \phi(x_k, \ldots, x_2, z, \bar{y})). \end{array}$$

The operators $\overline{\Delta}_h$ and Δ_v can be defined in a similar manner. As earlier, it is straightforward to see that

—$\Delta_h \phi(\bar{x}, \bar{y})$ holds if and only if $\phi(\bar{x}', \bar{y})$ holds, where $\bar{x}'$ is the lexicographical successor of $\bar{x}$, and
—$\overline{\Delta}_h \phi(\bar{x}, \bar{y})$ holds if and only if $\phi(\bar{x}', \bar{y})$ holds, where $\bar{x}$ is the lexicographical successor of $\bar{x}'$, and
—$\Delta_v \phi(\bar{x}, \bar{y})$ holds if and only if $\phi(\bar{x}, \bar{y}')$ holds, where $\bar{y}'$ is the lexicographical successor of $\bar{y}$.

This completes the definition of the space-time grid structure and thus completes our proof of Theorem 3.2. □

Next we recall a result by Grandjean stating that k-variable spectra, even if we use function symbols, can be computed effectively.

THEOREM 3.3 ([GRANDJEAN 1984; GRANDJEAN AND OLIVE 2004]). *For every integer $k \geq 1$,* F-SPEC$_k \subseteq$ NTIME$[n^2 2^{kn}]$.

Combining Theorems 3.2 and 3.3, we obtain the following hierarchy.

COROLLARY 3.4. *For every integer $k \geq 3$,* SPEC$_k \subsetneq$ SPEC$_{2k+3}$.

PROOF. The strict inclusion follows from

$$\text{SPEC}_k \subseteq \text{NTIME}[n^2 2^{kn}] \subsetneq \text{NTIME}[2^{(k+1)n}] \subseteq \text{SPEC}_{2(k+1)+1} = \text{SPEC}_{2k+3}.$$

The first inclusion follows from Theorem 3.3 and the third from Theorem 3.2. The second strict inclusion follows from Cook's nondeterministic time hierarchy theorem [Cook 1973; Arora and Barak 2009, Theorem 3.2]. □

The following corollary shows that to settle Asser's conjecture, it is sufficient to consider sentences using three variables and binary relations.

Define the following class:

$$\text{Co-SPEC}_3^{bin} := \left\{ \mathbb{N}^+ - S \;\middle|\; \begin{array}{l} S = \text{SPEC}(\phi) \text{ and } \phi \text{ uses only} \\ \text{three variables and binary relations} \end{array} \right\}.$$

COROLLARY 3.5. NE = CO-NE *if and only if* Co-SPEC$_3^{bin} \subseteq$ SPEC.

PROOF. The "only if" direction is trivial. The "if" direction is as follows. Suppose that Co-SPEC$_3^{bin} \subseteq$ SPEC. Since NTIME$[2^n] \subseteq$ SPEC$_3$ (and uses only binary relations), this means that for every $A \in$ NTIME$[2^n]$, the complement $\mathbb{N}^+ - A \in$ SPEC and hence also $\mathbb{N}^+ - A \in$ NE. By padding the argument, this implies that for every set $A \in$ NE, the complement $\mathbb{N}^+ - A$ also belongs to NE. □

To end this section, we present a slightly weaker result of Theorem 3.3—that is, SPEC$_k \subseteq$ NTIME$[n^2 2^{kn}]$, which is already sufficient to yield the hierarchy in Corollary 3.4. First we show the following normalization of first-order logic with k variables.

PROPOSITION 3.6 (NORMALIZATION OF FIRST-ORDER LOGIC WITH k VARIABLES). *Each first-order sentence ϕ with at most k distinct variables $\bar{x} = (x_1, \ldots, x_k)$ is equivalent to an existential second-order sentence of the form $\Phi := \exists R_1 \cdots \exists R_m\, \phi'$, where each R_i is a relation symbol of arity $\leq k$ and ϕ' is a conjunction of first-order sentences with variables $\bar{x} = (x_1, \ldots, x_k)$ of either of the following forms (1) and (2):*

(1) $\forall x_1 \cdots \forall x_{k-1}\, \forall x_k\, \psi(x_1, \ldots, x_k)$,
(2) $\forall x_1 \cdots \forall x_{k-1}\, \exists x_k\, \psi(x_1, \ldots, x_k)$,

where $\psi(x_1, \ldots, x_k)$ is a quantifier-free formula in disjunctive normal form.

PROOF. First we assume that all negations in ϕ are pushed inside to the atomic formulae.

We associate each subformula $\theta(v_1, \ldots, v_q)$ of ϕ, where $0 \le q \le k$ and each $v_i \in \bar{x}$, including the sentence ϕ, with a new relation symbol R_θ of arity q. The relation symbol R_θ intuitively represents θ. Note that a relation symbol of arity 0 is a Boolean variable that can be either true or false.

The formula ϕ' is the conjunction of the atomic relation R_ϕ of arity 0 and the formula δ_θ corresponding to subformula $\theta(v_1, \ldots, v_q)$ of ϕ defined inductively as follows:

—If θ is a negation of an atomic formula $S(v_1, \ldots, v_q)$, then

$$\delta_\theta \ := \ \forall v_1 \cdots \forall v_q \ R_\theta(v_1, \ldots, v_q) \Leftrightarrow \neg S(v_1, \ldots, v_q).$$

—If θ is of the form $\theta_1 \circledast \theta_2$, with free variables $v_1, \ldots, v_q$, where $\circledast \in \{\wedge, \vee\}$, then

$$\delta_\theta \ := \ \forall v_1 \cdots \forall v_q \ R_\theta(v_1, \ldots, v_q) \Leftrightarrow R_{\theta_1}(v_1, \ldots, v_q) \circledast R_{\theta_2}(v_1, \ldots, v_q).$$

Note that if θ has no free variable, then δ_θ is $R_\theta \Leftrightarrow R_{\theta_1} \circledast R_{\theta_2}$.

—If θ is $\forall v_q \ \theta'(v_1, \ldots, v_{q-1}, v_q)$, then

$$\delta_\theta \ := \ \forall v_1 \cdots \forall v_{q-1} \ R_\theta(v_1, \ldots, v_{q-1}) \Leftrightarrow \forall v_q \ R_{\theta'}(v_1, \ldots, v_q),$$

which is equivalent to

$$\begin{aligned} \delta_\theta \ := \ & \big(\forall v_1 \cdots \forall v_{q-1} \forall v_q \ R_\theta(v_1, \ldots, v_{q-1}) \Rightarrow R_{\theta'}(v_1, \ldots, v_q)\big) \ \wedge \\ & \big(\forall v_1 \cdots \forall v_{q-1} \exists v_q \ R_{\theta'}(v_1, \ldots, v_q) \Rightarrow R_\theta(v_1, \ldots, v_{q-1})\big). \end{aligned}$$

—If θ is $\exists v_q \ \theta'(v_1, \ldots, v_{q-1}, v_q)$, then

$$\delta_\theta \ := \ \forall v_1 \cdots \forall v_{q-1} \ R_\theta(v_1, \ldots, v_{q-1}) \Leftrightarrow \exists v_q \ R_{\theta'}(v_1, \ldots, v_q),$$

which is equivalent to

$$\begin{aligned} \delta_\theta \ := \ & \big(\forall v_1 \cdots \forall v_{q-1} \exists v_q \ R_\theta(v_1, \ldots, v_{q-1}) \Rightarrow R_{\theta'}(v_1, \ldots, v_q)\big) \ \wedge \\ & \big(\forall v_1 \cdots \forall v_{q-1} \forall v_q \ R_{\theta'}(v_1, \ldots, v_q) \Rightarrow R_\theta(v_1, \ldots, v_{q-1})\big). \end{aligned}$$

Note that in the preceding definition, if θ is an atomic formula, then R_θ is θ itself.

Written formally,

$$\Phi \ := \ \exists R_1 \ \cdots \ \exists R_m \ R_\phi \ \wedge \ \bigwedge_\theta \delta_\theta,$$

where $R_1, \ldots, R_m$ are all of the R_θ's and θ spans over all subformulae of ϕ. It is straightforward to see that Φ and ϕ are equivalent. □

The following complexity result is an easy consequence of the normalization lemma.

COROLLARY 3.7. *For every positive integer k,* $\text{SPEC}_k \subseteq \text{NTIME}[2^{kn}n^2]$.

PROOF. By the preceding lemma, each first-order sentence ϕ using k variables is equivalent to the normalized formula $\Phi := \exists R_1 \cdots \exists R_m \ \phi'$. By our construction, the quantification depth of ϕ' is k. Hence, on the domain $[N]$, where $N = \Theta(2^n)$, one can obtain a propositional Boolean formula $F_{\phi,N}$ with size $O(N^k)$[2] such that $N \in \text{SPEC}(\phi)$ if and only if $F_{\phi,N}$ is satisfiable.

It is well known that the satisfiability of problem of a propositional Boolean formula F of size ℓ with variables p_i of indices $i \le \ell$, hence, of total length $|F| = O(\ell \log \ell)$ (in a fixed finite alphabet), can be solved in time $O(\ell \log^2 \ell)$ on a nondeterministic Turing machine. We present it here in our specific case, where as a straightforward consequence of Proposition 3.6, the Boolean formula $F_{\phi,N}$ so obtained is a conjunction

[2]The size of a propositional Boolean formula is the total sum of the number of appearances of each atom.

of DNF formulae (i.e., of the form $F_{\phi,N} : C_1 \wedge \cdots C_m$) and each C_i is a DNF formula. It is easy to see that the satisfiability problem of a formula in such a form can be decided by the following nondeterministic algorithm:

—For each conjunct C_i, choose (nondeterministically) a disjunct γ_i of C_i. Note that γ_i is a conjunction of literals.
—Check deterministically whether the conjunction $G := \gamma_1 \wedge \cdots \wedge \gamma_m$, which is a conjunction $\ell_1 \wedge \cdots \wedge \ell_q$ of literals, is satisfiable. This can be done by sorting the list of literals $\ell_1, \ldots, \ell_q$ of G in lexicographical order and checking that the sorted list contains no pair of contiguous contradictory literals $p, \neg p$.

It is a folklore result that a list of nonempty words $w_1, \ldots, w_q$ can be sorted in lexicographical order on a multitape Turing machine in $O(\lambda \log \lambda)$, where $\lambda = |w_1| + \cdots + |w_q|$. Here we have $\lambda = |G| \leq |F_{\phi,N}| = O(\ell \log \ell)$. Altogether, it takes $O(\ell \log^2 \ell)$ time. □

4. A FINER HIERARCHY

In this section, we present a finer hierarchy of the spectra: for every integer $k \geq 3$, $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$. The outline of the proof follows the one in the previous section.

THEOREM 4.1. *For every integer $k \geq 2$,* $\mathrm{NTIME}[2^{(k+\frac{1}{2})n}] \subseteq \textsc{Spec}_{2k+2}$.

PROOF. We follow the outline of Proposition 3.1 and Theorem 3.2. Now, M is a nondeterministic Turing machine that accepts $\mathsf{BINARY}(N)$ in time $N^k R$ and space $N^k R$, where $R = \lfloor \sqrt{N-1} \rfloor$. The space-time diagram of the computation of M is then depicted as an $[N^k \cdot R] \times [N^k \cdot R]$ grid.

Each point in the $[N^k \cdot R] \times [N^k \cdot R]$ grid can be identified as a point in $[N]^k \times [R] \times [N]^k \times [R]$. By the converse of the pairing function $(r) \mapsto (\pi_x(r), \pi_y(r))$, where $\pi_x(r) = r \bmod R = r_1$, and $\pi_y(r) = \lfloor (r/R) \rfloor = r_2$, each point in $((\bar{x}, r_1), (\bar{y}, r_2)) \in [N]^k \times [R] \times [N]^k \times [R]$ can be represented as $(\bar{x}, \bar{y}, r) \in [N]^k \times [N]^k \times [N]$, where $r = r_1 + r_2 R$.

So the computation of M can be viewed as labeling of the point $(\bar{x}, \bar{y}, r) \in [N]^k \times [N]^k \times [N]$. The only difference now is that we need to define the shift operators Δ_h^r, $\overline{\Delta}_h^r$, and Δ_v^r, the analog of the shift operators Δ_h, $\overline{\Delta}_h$, and Δ_v, respectively, in the proof of Theorem 3.2.

As earlier, we define the order $<$, minimum MIN, maximum MAX, and the induced successor relation SUC. We also define the following relations:

—$\mathsf{ADD}(x, y, z)$, which holds if and only if $x + y = z$:

$$\forall x \forall y \forall z \left(\begin{array}{l} \mathsf{ADD}(x, y, z) \;\Leftrightarrow \\ \left(\begin{array}{l} (\mathsf{MIN}(y) \wedge x = z) \vee \\ (\exists y' \, \exists z' \; \mathsf{SUC}(y', y) \wedge \mathsf{SUC}(z', z) \wedge \mathsf{ADD}(x, y', z')) \end{array} \right) \end{array} \right).$$

—$\mathsf{MUL}(x, y, z)$, which holds if and only if $xy = z$:

$$\forall x \forall y \forall z \left(\begin{array}{l} \mathsf{MUL}(x, y, z) \;\Leftrightarrow \\ \left(\begin{array}{l} (\mathsf{MIN}(y) \wedge \mathsf{MIN}(z)) \vee \\ \exists y' \, \exists z' \; (\mathsf{SUC}(y', y) \wedge \mathsf{MUL}(x, y', z') \wedge \mathsf{ADD}(z', x, z) \end{array} \right) \end{array} \right).$$

—$\mathsf{IS\text{-}R}(x)$, which holds if and only if $x = R$

$$\forall x \; (\mathsf{IS\text{-}R}(x) \;\Leftrightarrow\; (\exists y \; \mathsf{MUL}(x, x, y) \wedge \neg \exists x' \exists y' \; x' > x \wedge \mathsf{MUL}(x', x', y'))).$$

—$\mathsf{LESS\text{-}R}(x)$, which holds if and only if $x < R$:

$$\forall x \; (\mathsf{LESS\text{-}R}(x) \;\Leftrightarrow\; \exists y \; y > x \wedge \mathsf{IS\text{-}R}(y)).$$

—LESS-R2(x), which holds if and only if $x < R^2$:

$$\forall x \ \big(\,\mathsf{LESS\text{-}R2}(x) \ \Leftrightarrow \ \exists y \ \exists z \ \mathsf{IS\text{-}R}(y) \ \wedge \ \mathsf{MUL}(y, y, z) \ \wedge \ x < z\,\big).$$

—PROJECT(r, x, y), which holds if and only if $x = \pi_x(r) = r \bmod R$ and $y = \pi_y(r) = \lfloor r/R \rfloor$:

$$\forall r \forall x \forall y \left(\begin{array}{l} \mathsf{PROJECT}(r, x, y) \ \Leftrightarrow \\ \left(\begin{array}{l} \mathsf{LESS\text{-}R2}(r) \wedge \mathsf{LESS\text{-}R}(x) \wedge \mathsf{LESS\text{-}R}(y) \wedge \\ \exists z \ \exists z' \ (\mathsf{IS\text{-}R}(z') \wedge \mathsf{MUL}(y, z', z) \wedge \mathsf{ADD}(x, z, r) \end{array} \right) \end{array} \right).$$

Using the preceding relations, it is straightforward to write the following definitions as first-order axioms using at most five variables:

—Cyclic successor in $[R]$:
RCYC(x, y) if and only if $x, y \in [R]$, and either $y = x + 1$ or $x = R - 1$ and $y = 0$.
—Horizontal successor in $[R^2]$:
SUCX(r, r') if and only if $r, r' \in [R^2]$, $\pi_y(r) = \pi_y(r')$, and RCYC$(\pi_x(r), \pi_x(r'))$.
—Vertical successor in $[R^2]$:
SUCY(r, r') if and only if $r, r' \in [R^2]$, $\pi_x(r) = \pi_x(r')$, and RCYC$(\pi_y(r), \pi_y(r'))$.
—Horizontal minimum in $[R^2]$:
MINX(r) if and only if $r \in [R^2]$ and $\pi_x(r) = 0$.
—Vertical minimum in $[R^2]$:
MINY(r) if and only if $r \in [R^2]$ and $\pi_y(r) = 0$.

All of the preceding definitions use at most five variables, which is $\leq 2k + 2$, for each integer $k \geq 2$.

The operators $\Delta_h^r \phi$, $\overline{\Delta}_h^r \phi$, and $\Delta_v^r \phi$ are defined as follows:

$$\begin{aligned}
\Delta_h^r \phi(\bar{x}, \bar{y}, r) &:= \forall z \left(\mathsf{SUCX}(r, z) \Rightarrow \left(\begin{array}{lcl} (\mathsf{MINX}(z) & \wedge & \Delta_h \phi(\bar{x}, \bar{y}, z)) \vee \\ (\neg \mathsf{MINX}(z) & \wedge & \phi(\bar{x}, \bar{y}, z))) \end{array} \right) \right) \\
\overline{\Delta}_h^r \phi(\bar{x}, \bar{y}, r) &:= \forall z \left(\mathsf{SUCX}(z, r) \Rightarrow \left(\begin{array}{lcl} (\mathsf{MINX}(r) & \wedge & \overline{\Delta}_h \phi(\bar{x}, \bar{y}, z)) \vee \\ (\neg \mathsf{MINX}(r) & \wedge & \phi(\bar{x}, \bar{y}, z))) \end{array} \right) \right) \\
\Delta_v^r \phi(\bar{x}, \bar{y}, r) &:= \forall z \left(\mathsf{SUCY}(r, z) \Rightarrow \left(\begin{array}{lcl} (\mathsf{MINY}(z) & \wedge & \Delta_v \phi(\bar{x}, \bar{y}, z)) \vee \\ (\neg \mathsf{MINY}(z) & \wedge & \phi(\bar{x}, \bar{y}, z))) \end{array} \right) \right),
\end{aligned}$$

where Δ_h is to access the successor of $\bar{x}$, $\overline{\Delta}_h$ the predecessor of $\bar{x}$, and Δ_v the successor of $\bar{y}$. They are all defined just like in the proof of Theorem 3.2. This completes our proof of Theorem 4.1. □

Now, combining both Theorems 4.1 and 3.3, as well as the argument in the proof of Corollary 3.4, we obtain that

$$\textsc{Spec}_k \subseteq \text{NTIME}[n^2 2^{kn}] \subsetneq \text{NTIME}[2^{(k+\frac{1}{2})n}] \subseteq \textsc{Spec}_{2k+2},$$

hence establishing the following hierarchy.

Corollary 4.2. *For every integer $k \geq 3$, $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$.*

5. TRANSLATING OUR RESULTS TO CLASSES NP AND SO∃

In this section, we show how our results can be translated into relations between the class NP and the class of existential second-order sentences SO∃. We provide a brief review of their definitions here. For more details, we refer the reader to Immerman's textbook [Immerman 1999].

Let SO$\exists$ denote the class of existential second-order sentences. A sentence $\Phi \in$ SO$\exists$ defines a class of structures $\{\mathfrak{A} \mid \mathfrak{A} \models \Phi\}$. A celebrated result of Fagin states that SO$\exists$ = NP, where the input to the NP Turing machine is the binary encoding of the structures.

Let SO$\exists$(var k) be the class SO$\exists$ where the first-order sentences use only up to k variables. Now, Theorems 3.2 and 4.1 can be respectively rewritten as

$$\text{For any integer } k \geq 1, \quad \mathrm{NTIME}[n^k] \subseteq \mathrm{SO}\exists(\mathrm{var}\ 2k+1) \tag{1}$$

$$\text{For any integer } k \geq 2, \quad \mathrm{NTIME}[n^{k+1/2}] \subseteq \mathrm{SO}\exists(\mathrm{var}\ 2k+2). \tag{2}$$

Indeed, let M be a nondeterministic Turing machine accepting a binary language L within time $O(n^k)$, where n is the length of the input string $w = w_0 \dots w_{n-1} \in \{0,1\}^*$. The input can be viewed as a structure over $[n]$ with vocabulary the binary successor relation SUC and the unary predicate S, where $S(x)$ holds if and only if $w_x = 1$.

The formula Φ constructed in the proof of Theorem 3.2 (resp. Theorem 4.1) can be viewed as an SO$\exists$(var $2k+1$) (resp. SO$\exists$(var $2k+2$)) formula, where the predicates SYMBOL_a^i and STATE_q^i, as well as DOUBLE, HALF, DIV, BIT, and so on, are existentially quantified.

On the other hand, Theorem 3.3 can be rewritten as

$$\mathrm{SO}\exists(\mathrm{var}\ k) \subseteq \mathrm{NTIME}[n^k \log^2 n]. \tag{3}$$

Equations (3) and (2) then yield the chain of inclusions:

$$\mathrm{SO}\exists(\mathrm{var}\ k) \ \subseteq\ \mathrm{NTIME}[n^k \log^2 n] \ \subsetneq\ \mathrm{NTIME}[n^{k+1/2}] \ \subseteq\ \mathrm{SO}\exists(\mathrm{var}\ 2k+2),$$

and hence SO$\exists$(var k) $\subsetneq$ SO$\exists$(var $2k+2$) for each $k \geq 3$.

6. CONCLUDING REMARKS

In this article, we present two results that we believe contribute to our understanding of the spectra problem. The first is that there is an infinite hierarchy of first-order spectra based on the number of variables: $\textsc{Spec}_k \subsetneq \textsc{Spec}_{2k+2}$. The proof is based on tight relationships between the class NE and first-order spectra $\textsc{Spec}$.

The second result is that to settle Asser's conjecture, it is sufficient to consider sentences using three variables and binary relations. This seems to be the furthest we can go. As mentioned in Section 2, we recently showed that the class of spectra of two-variable logic with counting quantifiers is exactly the class of semilinear sets and closed under complement [Kopczyński and Tan 2015].

ACKNOWLEDGMENTS

We would like to thank Etienne Grandjean for his careful reading and providing most of the literature pointers in Section 2, as well as suggesting Proposition 3.6 and Corollary 3.7. His extensive comments have greatly guided and helped us in improving this article. We also thank Ron Fagin for some discussions on related results and the anonymous reviewer for his careful review.

REFERENCES

S. Arora and B. Barak. 2009. *Computational Complexity: A Modern Approach*. Cambridge University Press.

G. Asser. 1955. Das repräsentenproblem in prädikatenkalkül der ersten Stufe mit identität. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik* 1, 252–263.

J. Chen, X. Huang, I. A. Kanj, and G. Xia. 2004. Linear FPT reductions and computational lower bounds. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*. 212–221.

J. Chen, X. Huang, I. A. Kanj, and G. Xia. 2006. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences* 72, 8, 1346–1367.

S. Cook. 1973. A hierarchy for nondeterministic time complexity. *Journal of Computer and System Sciences* 7, 4, 343–353.

A. Durand, N. D. Jones, J. A. Makowsky, and M. More. 2012. Fifty years of the spectrum problem: Survey and new results. *Bulletin of Symbolic Logic* 18, 4, 505–553.

R. Fagin. 1973. *Contributions to the Model Theory of Finite Structures*. Ph.D. Dissertation. University of California, Berkeley.

R. Fagin. 1974. Generalized first-order spectra and polynomial-time recognizable sets. In *Proceedings of SIAM-AMS Complexity of Computation*. 27–41.

R. Fagin. 1975. A spectrum hierarchy. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik* 21, 123–134.

R. Fagin. 1993. Finite-model theory: A personal perspective. *Theoretical Computer Science* 116, 1–2, 3–31.

E. Grandjean. 1984. The spectra of first-order sentences and computational complexity. *SIAM Journal on Computing* 13, 2, 356–373.

E. Grandjean. 1985. Universal quantifiers and time complexity of random access machines. *Mathematical Systems Theory* 18, 2, 171–187.

E. Grandjean. 1990. First-order spectra with one variable. *Journal of Computer and System Sciences* 40, 2, 136–153.

E. Grandjean and F. Olive. 2004. Graph properties checkable in linear time in the number of vertices. *Journal of Computer and System Sciences* 68, 3, 546–597.

N. Immerman. 1999. *Descriptive Complexity*. Springer.

R. Impagliazzo and R. Paturi. 1999. Complexity of k-SAT. In *Proceedings of the IEEE Conference on Computational Complexity*.

N. Jones and A. Selman. 1974. Turing machines and the spectra of first-order formulas. *Journal of Symbolic Logic* 39, 139–150.

E. Kopczyński and T. Tan. 2015. Regular graphs and the spectra of two-variable logic with counting. To appear in *SIAM Journal on Computing.*

J. Lynch. 1982. Complexity classes and theories of finite models. *Mathematical Systems Theory* 15, 2, 127–144.

C. H. Papadimitriou. 1994. *Computational Complexity*. Addison-Wesley.

P. Pudlák. 1975. The observational predicate calculus and complexity of computations. *Commentationes Mathematicae Universitatis Carolinae* 16, 395–398.

H. Scholz. 1952. Ein ungelöstes problem in der symbolischen logic. *Journal of Symbolic Logic* 17, 160.

Received April 2014; revised January 2015; accepted February 2015