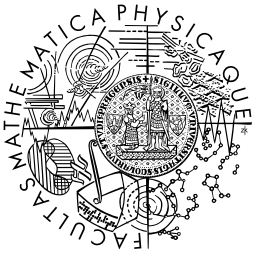

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

BACHELOR THESIS

Ondřej Ježil

Spectrum Problem

Department of Algebra

Supervisor of the bachelor thesis: prof. RNDr. Jan Krajíček, DrSc.

Study programme: Mathematics for Information Technologies

Study branch: Mathematics

Prague 2020

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ....

Author's signature

I dedicate this thesis to my parents. I could not have completed it without their love and support. I am obliged to my supervisor, prof. RNDr. Jan Krajíček, DrSC., for sharing his knowledge, thoroughly commenting on this thesis and also for introducing me to this marvelous topic.

Title: Spectrum Problem

Author: Ondřej Ježil

Department: Department of Algebra

Supervisor: prof. RNDr. Jan Krajíček, DrSc., Department of Algebra

Abstract: We study spectra of first-order sentences. After providing some interesting examples of spectra we show that the class of spectra is closed under some simple set-theoretic and algebraic operations. We then define a new class of definable operations generalizing the earlier constructions. Our main result is that the class of these operations is, in a suitable technical sense, closed under a form of iteration. This in conjunction with Cobham's characterisation of **FP** offers a new proof of Fagin's theorem and also of the Jones-Selman characterisation of spectra as **NE** sets.

Keywords: Scholz's problem, spectrum, Asser's problem, generalized spectrum, Fagin's theorem, Cobham's theorem

Contents

Introduction

Asser's Spectrum Problem is one of the oldest open questions that directly relate to the famous **P** vs. **NP** problem of computational complexity. Bordering between mathematical logic and complexity theory it asks whether spectra of first-order sentences are closed under complementation. Spectra are simply the sets of cardinalities of finite models of some first-order sentence.

Scholz introduced spectra in [Sch52] and asked for their characterization. This became known as Scholz's spectrum problem. It is generally considered to be solved since spectra have been characterized in terms of computational complexity [Fag74] [JS74]. Even though new answers to it can still emerge. In [Ass55] Asser responded to Scholz's paper and asked the question of whether the spectra are closed under complementation.

The study of spectra leads to many interesting ideas which helped to form at least two subfields of mathematical logic. Namely finite model theory and descriptive complexity theory [DJMM12].

We will first recall necessary definitions and facts and introduce some notation, both from logic and complexity theory. We will then present elementary results on spectra which are not too hard to prove without any special theory. In Chapter 3 we define a new notion of Σ^1_1-definable functions and use them to give a new proof of Fagin's theorem originally proved in [Fag74]. The characterization of spectra as **NE** sets is then a simple corollary.

The fundamental issue in studying spectra is the underlying connection between finite models and complexity theory. Instead of some class of finite models of some sentence we can just consider the set of their binary codes. Now a class of finite models suddenly becomes a binary language. On the other hand, binary words can be treated as a special class of word structures. See [GKL$^+$07] for more detailed explanation.

The survey article [DJMM12] is an extensive report on both historical and contemporary directions of studying spectra and we recommend it to a reader interested in other approaches.

1. Preliminaries

1.1 Logic

We will assume basic knowledge of first-order logic and model theory. We point interested reader to [vdD10] and [Mar06] for a nice introduction into those topics.

Let us start with some notation and terminology. We will use the word vocabulary for what is in mathematical logic more commonly known as a language. Reason being that language is also a term used in complexity theory for sets of words over an alphabet.

Now some comments about our notations for formulas. For some unary relational symbol U and variable x we will use the notation $U(x)$ and $x \in U$ interchangeably. We will also use the quantifier $(\exists! x)$ meaning "there exists a unique". Also the "bounded quantifiers" $(\forall x \in U)$ and $(\exists x \in U)$ which can be understood as abbreviations for $(\forall x)(x \in U \to \dots)$ and $(\exists x)(x \in U \wedge \dots)$ respectively. In linearly ordered structures we will use the quantifiers $(\forall x < y)$, $(\exists x < y)$, symbols $>, \leq, \geq$ with their obvious interpretation and $\max R$ to denote the maximum of a non-empty unary relation R. Any such notation can be rewritten to an actual formula. Sometimes we will treat n-ary relational symbols with a fixed variable i as unary symbols by using the notation $x \in M_i$ for $M(i, x)$, where M_- is the actual relational symbol.

For convenience we will only consider relational vocabularies. We will however not lose the rich supply of examples from algebra. Every binary functional symbol can be replaced with $(n+1)$-ary relational symbol representing the graph of the functional one. This restriction is traditional in finite model theory as it makes the notion of a substructure more manageable.

Definition 1.1.1 (Spectrum)**.** Let φ be a first-order formula in a purely relational vocabulary τ. We define the spectrum of τ-sentence φ as

$$\mathrm{Spec}_\tau(\varphi) = \{|A|; \mathcal{A} \text{ is a finite } \tau\text{-structure and } \mathcal{A} \models \varphi\}. \tag{1.1.1}$$

Formally, there is no way we can tell if said φ is a τ-formula, or formula of some subset of this vocabulary which contains all non-logical symbols occurring in φ. E.g. $\varphi_1 = (x = y)$ is both a $\{<\}$-formula and a $\varnothing$-formula. But since φ does not "talk" about other relational symbols in τ, their interpretations of any model of the smaller vocabulary can be chosen arbitrarily to expand it to a τ-structure. Conversely, we can just forget these relations and take a reduct of the model to the smaller vocabulary.

That means we can simply write

$$\mathrm{Spec}(\varphi) = \{|A|; \mathcal{A} \text{ is a finite structure and } \mathcal{A} \models \varphi\}. \tag{1.1.2}$$

Definition 1.1.2 (Spec)**.** We denote the set of all spectra Spec.

Spec will turn out to be a complexity class, which we will characterize in terms of a more classical complexity theoretic notion. The corollary of Fagin's theorem will be that Spec is exactly the set of all languages accepted in nondeterministic

exponential-time. Specifically in NTime($2^{\mathcal{O}(n)}$). We will give a new proof of this influential theorem in Chapter 3.

In Chapter 2 we give a few examples of spectra.

It turned out to be important to look at an analogue of spectra for a logic different from the first-order one to understand the usual spectra. Namely we will turn our attention to existential second-order logic denoted Σ^1_1.

Definition 1.1.3 (Σ^1_1-formulas)**.** Let τ be a vocabulary. **Σ^1_1-formulas** in the vocabulary τ are of the form

$$(\exists_{c_1} X_1)(\exists_{c_2} X_2)\dots(\exists_{c_n} X_n)\varphi(\bar{x}), \tag{1.1.3}$$

where φ is a first-order formula and X_i is a c_i-ary relational symbol not in τ for each $i \in \{1,\dots,n\}$. In this context we call relational symbols not in τ second-order variables. The notation $\psi(Y_1,\dots,Y_k,x_1,\dots,x_l)$ for a Σ^1_1-formula means that its unquantified (free) second-order variables are among $\{Y_1,\dots,Y_k\}$ and its free first-order variables are among $\{x_1,\dots,x_l\}$.

Σ^1_1-formula without both free second-order variables and free first-order variables is called a **Σ^1_1-sentence**.

The notation $(\exists_c Z)$ is nonstandard, usually the arity in the subscript is omitted and is simply deduced from the type of the relational symbol. The reason for its inclusion is that in later chapters we will encounter Σ^1_1-formulas with a great number of quantified relational symbols of different arities. Using for example different fonts to distinguish arity of relational symbols would become unwieldy.

Note that Σ^1_1-formulas are closed (up to logical equivalence) under conjunction, disjunction and first-order quantifications. We will use this heavily in later chapters.

Definition 1.1.4 (Validity of Σ^1_1-formulas)**.** Let $\varphi(X_1,\dots,X_k;x_1,\dots,x_l)$ be a Σ^1_1-formula in the vocabulary τ. Let $\mathcal{A}$ be a τ-structure, let $R_1,\dots,R_k$ be relations on A with arities corresponding to X_i's and let $a_1,\dots,a_l \in A$. Then we define the **validity of φ in $\mathcal{A}$ with assignment** $R_1,\dots,R_k,a_1,\dots,a_l$, denoted $\mathcal{A} \models \varphi(R_1,\dots,R_k,a_1,\dots,a_l))$, as follows. For $\varphi(X_1,\dots,X_k,x_1,\dots,x_l)$ without second-order quantification we define the satisfaction relation as in Tarski's definition.

For $\varphi = (\exists_{c_1} Z_1)\dots(\exists_{c_n} Z_n)\varphi_0(Z_1,\dots,Z_n,X_1,\dots,X_k,x_1,\dots,x_l)$ we define

$$\mathcal{A} \models \varphi(R_1,\dots,R_k,a_1,\dots,a_l) \tag{1.1.4}$$

iff there exists interpretations of Z_i called B_i for all $i \in \{1,\dots,n\}$ such that

$$(\mathcal{A},B_1,\dots,B_n,R_1,\dots,R_k,a_1,\dots,a_l) \models \varphi_0(B_1,\dots,B_n,R_1,\dots,R_k,\bar{a}). \tag{1.1.5}$$

We call an expansion of $\mathcal{A}$ containing a satisfying interpretation for each existentially quantified symbols in φ an **witnessing expansion** of $\mathcal{A}$. If φ is a Σ^1_1-sentence and $\mathcal{A} \models \varphi$ then we call $\mathcal{A}$ **a model of** φ.

Definition 1.1.5. Let $\varphi = (\exists_{c_1} Z_1)\dots(\exists_{c_n} Z_n)\varphi_0(Z_1,\dots,Z_n,\bar{Y},\bar{x})$ be a Σ^1_1-formula. We define its first-order part as φ_0 and denote it by φ^{FO}.

For example $(\exists_2 R)R(x,y)$ is a Σ^1_1-formula, $(\exists_2 R)(\exists y)(\forall x)(R(x,y))$ is a Σ^1_1-sentence. Many examples of models of an Σ^1_1-sentence come from algebra. Namely if you consider a conjunction of axioms of groups (their relational version), by existentially quantifying a unary relation, different from the singleton $\{1\}$ and the whole group, that is closed under multiplication, inverses and conjugation by every element in the group, the resulting models are exactly non-simple groups. A similar construction can be carried out for local rings, solvable groups of fixed order and other notions from algebra that depend on the existence of some subset with special properties.

Definition 1.1.6 (Generalized spectra)**.** Let φ be a Σ^1_1-sentence in some vocabulary τ. We define the **generalized spectrum** of φ

$$\mathrm{GenSpec}_\tau(\varphi) := \{\mathcal{A}; \mathcal{A} \models \varphi, |A| < \aleph_0\}. \tag{1.1.6}$$

In other words it is the class of all finite models of φ. We call φ **a defining sentence of the generalized spectrum.** As opposed to (regular) spectra, here different choices of vocabulary result in different generalized spectra. However, when the vocabulary is clear from the context we will simply write $\mathrm{GenSpec}(\varphi)$.

We denote GenSpec the class of all generalized spectra. After some binary encoding of finite structures is fixed we can instead of each generalized spectrum consider the set of binary codes of its members. Fagin's theorem then characterizes GenSpec as those classes of finite structures whose set of codes is in **NP**.

An important connection to first-order spectra is that for a first-order sentence φ in some vocabulary τ if we quantify all the relational symbols in τ we get $\hat{\varphi} = (\exists_{a_1} R_1) \dots (\exists_{a_k} R_k)\varphi$. The generalized spectrum of $\hat{\varphi}$ contains exactly the structures which contain no relations with an universe of cardinality from $\mathrm{Spec}(\varphi)$. These structures will later correspond to unary encodings of numbers.

1.2 Cobham's characterization of FP

In this section we present the complexity class **FP** and Cobham's characterization of it. We will not introduce all the terminology from complexity theory and instead direct the interested reader to [AB09] for an extensive explanation of the subject.

Definition 1.2.1. FP denotes the class of functions, with arguments and values in the set of binary words, that are computable by a polynomial-time algorithm.

We will now present a theorem proved by Cobham [Cob65]. It provides a recursion theory style characterization of **FP**, without referring to any formal notion of machine or computation.

For $x \in \{0,1\}^*$ we denote its bit length by $|x|$.

Theorem 1.2.2 (Cobham [Cob65])
Define B as the smallest class of functions that:

1. contains

(a) successor functions $s_i : \{0,1\}^* \to \{0,1\}^*$, $s_i(x) = xi$, where $i \in \{0,1\}$,

(b) projections $\pi_i^n : (\{0,1\}^*)^n \to \{0,1\} : (x_1, \ldots, x_n) \mapsto x_i$, where $n \in \mathbb{N}^+, i \leq n$,

(c) binary function $x\#y := 2^{|x|\cdot|y|} = \underbrace{1000\ldots0}_{|x|\cdot|y| \text{ zeroes}}$, called the smash function,

2. is closed under composition,

3. is closed under limited recursion on notation, that is, for each $g, h_0, h_1 \in B$, where g is n-ary and h_0, h_1 $(n+2)$-ary. If for the $(n+1)$-ary function f defined as

$$f(x_1, \ldots, x_n, 0) := g(x_1, \ldots, x_n) \tag{1.2.1}$$
$$f(x_1, \ldots, x_n, s_i(x)) := h_i(x_1, \ldots, x_n, x, f(x)) \tag{1.2.2}$$

there exists $c \in \mathbb{N}^+$ such that for all $x_1, \ldots, x_n \in \{0,1\}^*$:

$$|f(x_1, \ldots, x_n, x)| \leq \max(|x_1|, \ldots, |x_n|, |x|, 2)^c, \tag{1.2.3}$$

then $f \in B$.

Then $B = \mathbf{FP}$.

We will use this characterization in Chapter 3 to prove Fagin's theorem in a way that completely avoids representing computations of machines by finite structures.

2. Elementary results on spectra

2.1 Interesting examples of spectra

In this section we provide a few interesting examples of spectra and we will start with some simple ones. One of the most basic examples of spectra are finite and cofinite sets. First we consider the sentence

$$\mathrm{card}_k = (\exists x_1)\dots(\exists x_k)((\bigwedge_{1\le i<j\le k} x_i \neq x_j) \wedge (\forall y)(\bigvee_{1\le i\le k} y = x_i)), \tag{2.1.1}$$

which states that the cardinality of any model of this sentence is exactly $k \in \mathbb{N}^+$. Any finite set $\{n_1, \dots, n_m\}$ can be defined by a disjunction of all $\mathrm{card}_{n_i}, i \in \{1, \dots, m\}$. Any cofinite set can be defined by the negation of such a sentence.

Moreover it can be shown that spectra of $\varnothing$-sentences are precisely those. First note that for spectra over the empty vocabulary we have $\mathrm{Spec}(\neg\varphi) = \mathbb{N}^+ \setminus \mathrm{Spec}(\varphi)$. (**!** This does not hold for all vocabularies. See section 2.2 for more details.) Namely, consider $\varnothing$-sentence φ such that both $\mathrm{Spec}(\varphi)$ and $\mathbb{N}^+ \setminus \mathrm{Spec}(\varphi)$ are infinite then by the compactness theorem and Löwenheim-Skolem theorem we have countable structures $\mathcal{A} \models \varphi$ and $\mathcal{B} \models \neg\varphi$, but those structures are necessarily isomorphic because they are just sets with no relations. A contradiction.

It follows from Fagin's theorem 3.3.1 that even many "complex" sets are spectra. Fagin's theorem uses logic notions but in this chapter we shall give examples of spectra that have mathematically natural constructions.

2.1.1 Spectra based on factors

Simple examples of spectra are the sets $k\mathbb{N}^+$ for some $k \in \mathbb{N}^+$. Those are exactly the spectra of the $\{X, Y, I\}$-sentences

$$\begin{aligned}\varphi_k :=& \mathrm{card}_k(X) \wedge (\forall x \in X)(\forall y \in Y)(\exists! z)(I(x,y,z)) \\ &\wedge (\forall z)(\exists! x \in X)(\exists! y \in Y)(I(x,y,z)),\end{aligned} \tag{2.1.2}$$

for some $k \in \mathbb{N}^+$, where

$$\begin{aligned}\mathrm{card}_k(X) :=& (\exists x_1 \in X)\dots(\exists x_k \in X)((\bigwedge_{1\le i<j\le k} x_i \neq x_k) \\ &\wedge (\forall y \in X)(\bigvee_{1\le i\le k} x_i = y))\end{aligned} \tag{2.1.3}$$

which is a sentence stating that X has cardinality k. The sentence φ_k states that I is a graph of a bijection between $X \times Y$ and the whole universe of the structure. It follows that the size of every finite model of φ_k is divisible by k, therefore $\mathrm{Spec}(\varphi_k) = k\mathbb{N}^+$.

More interesting are the sets of the form

$$\langle p_1, \dots, p_n\rangle_{(\mathbb{N}^+,\cdot)} = \{\prod_{j=1}^{n} p_j^{i_j}; i_j \in \mathbb{N}^+\}, \tag{2.1.4}$$

where the left side denotes closure under the operation in the monoid $(\mathbb{N}^+, \cdot)$. Here, instead of specifying which factors *should* be present in the elements of the sets, we specify which factors *can* be present.

These cannot be proved to be spectra by an analogous argument. If we include some ternary relation $I(x, y, z)$ in the sentence we can just state properties it needs to fulfill. This works if we want check if some mapping exists. However we need to check that every mapping fails to witness that some factor is outside of $\{p_1, \ldots, p_n\}$. There is no direct way to adapt the earlier simple argument.

Example 2.1.1
The sets $\langle p_1, \ldots, p_n \rangle_{(\mathbb{N}^+, \cdot)}$ are spectra.

Proof. We will use the fact that these sets are precisely the cardinalities of finite $\mathbb{Z}/(p_1 \ldots p_n)\mathbb{Z}$-modules. First notice that $\mathbb{Z}/k\mathbb{Z}$-modules are just abelian groups satisfying the equality

$$\underbrace{x + \cdots + x}_{k \text{ times}} = 0. \tag{2.1.5}$$

Now from (2.1.5) we have that every element of a finite $\mathbb{Z}/k\mathbb{Z}$-module M must have order dividing k.

From the classification of finitely generated abelian groups, proof of which can be found in [Rot10], we have that for M there exist primes $p'_1, \ldots, p'_m$ and natural numbers $i_1, \ldots, i_m$ such that $M \cong \bigoplus_{j=1}^{m} \mathbb{Z}/(p'_j)^{i_j}\mathbb{Z}$, therefore all $(p'_j)^{i_j}$ have to divide k. Also a module of every needed cardinality can be constructed as a $\bigoplus_{j=1}^{o}(\mathbb{Z}/p'_j\mathbb{Z})^{l_j}$ for some natural numbers $l_1, \ldots, l_o$.

Now we just need a sentence which axiomatizes $\mathbb{Z}/k\mathbb{Z}$-modules for $k := \prod p_i$. We consider the vocabulary $\tau = \{A, I, N\}$, where A is ternary, for the graph of addition, I is binary, for the graph of inverse, and N is unary, singling out the neutral element.

We construct the sentence mod_k as a conjunction of the universal closures of the following formulas.

$$\mathrm{const}_N = (\exists n)(N(n) \wedge (\forall m)(N(m) \to m = n)) \tag{2.1.6}$$
$$\mathrm{unfcn}_I = (\exists y)(I(x, y) \wedge (\forall z)(I(x, z) \to y = z)) \tag{2.1.7}$$
$$\mathrm{bifcn}_A = (\exists v)(A(x, y, v) \wedge (\forall w)(A(x, y, w) \to v = w)) \tag{2.1.8}$$
$$\mathrm{assoc} = (A(x, y, v) \wedge A(y, z, w)) \to (\forall u)(A(x, w, u) \leftrightarrow A(v, z, u)) \tag{2.1.9}$$
$$\mathrm{neutr} = N(n) \to (A(x, n, x) \wedge A(n, x, x)) \tag{2.1.10}$$
$$\mathrm{inver} = I(x, y) \to (\forall z)(A(x, y, z) \to N(z))) \tag{2.1.11}$$
$$\mathrm{commt} = A(x, y, z) \leftrightarrow A(y, x, z) \tag{2.1.12}$$
$$\mathrm{modul}_k = (A(x, x, x_2) \wedge A(x_2, x, x_3) \wedge \cdots \wedge A(x_{k-1}, x, x_k)) \to N(x_k) \tag{2.1.13}$$

Here (2.1.6)-(2.1.8) just state N, I, A are graphs of a constant, a unary function and a binary function respectively. The rest of the formulas are just the usual axioms of abelian groups rewritten using the graphs of the operations and the last formula is there to limit the models to $\mathbb{Z}/k\mathbb{Z}$-modules.

It follows that $\mathrm{Spec}(\mathrm{mod}_k) = \langle p_1, \ldots, p_k \rangle_{(\mathbb{N}^+, \cdot)}$. □

α	β	γ
β	γ	α
γ	α	β

α	β	γ
γ	α	β
β	γ	α

(α,α)	(β,β)	(γ,γ)
(β,γ)	(γ,α)	(α,β)
(γ,β)	(α,γ)	(β,α)

Figure 2.1: 2 mutually orthogonal latin squares over the of set of size 3 and their overlap.

2.1.2 Mutually orthogonal latin squares

In this section we will demonstrate how computationally unfeasible can individual spectra be by describing a specific spectrum for which it is unknown whether it contains number 10.

Latin square is a tuple $(A, \cdot)$, where $\cdot$ is a binary operation on a finite set A, which satisfies both that $(\forall x)(\forall z)(\exists y)(x \cdot y = z)$ and $(\forall y)(\forall z)(\exists x)(x \cdot y = z)$. That is if we look at the table of $\cdot$ we have that in every row there is each element of A exactly once and the same for every column.

Definition 2.1.2. Let $X = \{(A, \cdot_1), \ldots, (A, \cdot_k)\}$ be a set of latin squares over the same set A. We say that the latin squares in X are **mutually orthogonal** if the mapping $(x, y) \mapsto (x \cdot_1 y, \ldots, x \cdot_k y)$ is injective.

This is equivalent to the more visual condition (see Figure 2.1) that when you overlap all the tables of $\cdot_i$, for $i \in \{1, \ldots, k\}$, the resulting table contains no k-tuple twice. We will use the abbreviation **MOLS** for **mutually orthogonal latin squares**.

Mutually orthogonal latin squares are studied in combinatorics. There are many open questions surrounding them [CD01]. We will use this to demonstrate that membership of some number in a spectrum can be an open problem.

One of the open problems about latin squares asks what what is the maximum number of MOLS for each finite $n = |A|$. It is not hard to show that there cannot be more than $n - 1$ MOLS for a set of size n. And for $n = p^m$ for some prime number p and a natural number m it is not hard to show a general construction for achieving this upper bound using the theory of finite fields [Bos38]. We will call MOLS_k the set of all cardinalities of sets A that permit the existence k mutually orthogonal latin squares.

Example 2.1.3
MOLS_k is a spectrum for each $k \in \mathbb{N}^+$.

Proof. This can be shown by directly writing down the axioms for tables of $\cdot_1, \ldots, \cdot_k$. Again, we will introduce a couple of formulas and consider the sentence mols_k as a conjunction of the universal closures of those formulas. We will use the vocabulary $\{M_1, \ldots, M_k\}$ of ternary relational symbols representing the graphs of the individual operations. For each $i \in \{1, \ldots, k\}$ we put

$$\mathrm{bifcn}_{M_i} = (\exists v)(M_i(x, y, v) \wedge (\forall w)(M_i(x, y, w) \to v = w)) \tag{2.1.14}$$

$$\mathrm{ldiv}_{M_i} = (\exists x)(M_i(x, y, z) \wedge (\forall x')(M_i(x', y, z) \to x = x')) \tag{2.1.15}$$

$$\mathrm{rdiv}_{M_i} = (\exists y)(M_i(x, y, z) \wedge (\forall y')(M_i(x, y', z) \to y = y')) \tag{2.1.16}$$

The universal closures of these formulas just state that M_i is a graph of a binary function that satisfies both axioms of latin squares.

Now for the mutual orthogonality we put

$$\begin{aligned}\text{orth} = \bigwedge_{1\leq i\leq k} (M_i(x,y,z_i) \rightarrow \\ (\forall x')(\forall y')(\bigwedge_{1\leq i\leq k} M_i(x',y',z_i) \rightarrow (x=x' \wedge y=y'))).\end{aligned} \tag{2.1.17}$$

It is clear that models of mols_k are tables of k mutually orthogonal latin square operations. That is $\text{MOLS}_k = \text{Spec}(\text{mols}_k)$. □

MOLS_k always contains every prime power greater than $k-1$. What other elements are in these sets is a subject of active research. For example, it is unknown whether there exist three mutually orthogonal latin squares on a 10 element set [CD01]. In other words

$$10 \stackrel{?}{\in} \text{Spec}(\text{mols}_3) \tag{2.1.18}$$

is an open question.

2.2 Operations on spectra

A natural idea is to start with a few simple spectra and generate more complex ones using some operations that result in potentially new spectra.

2.2.1 Set operations on spectra

Lemma 2.2.1
Let $A, B \in \text{SPEC}$, then $A \cup B \in \text{SPEC}$ and $A \cap B \in \text{SPEC}$.

Proof. We assume that $A = \text{Spec}(\varphi_A)$ $B = \text{Spec}(\varphi_B)$. It is not hard to see that $\text{Spec}(\varphi_A \vee \varphi_B) = A \cup B$.

For $A \cap B$ we have to be a bit more careful. If we have $\mathcal{A} \models \varphi_A$ and $\mathcal{B} \models \varphi_B$ of size n, the sentence $\varphi_A \wedge \varphi_B$ might be unsatisfiable if the vocabularies of φ_A and φ_B are not disjoint. This can be fixed by constructing a sentence φ'_B by replacing each relational symbol in it by some symbol of the same arity which is not used in φ_A. Then $\text{Spec}(\varphi_A \wedge \varphi'_B) = A \cap B$, since you can construct each model of $\varphi_A \wedge \varphi'_B$ by superposing a model of φ_A on a model of φ_B of the same cardinality. On the other hand from each model of $\varphi_A \wedge \varphi'_B$ we can just take reducts to the vocabularies of φ_A and φ'_B. □

A natural question becomes whether or not is the class of spectra closed under complementation. That is whether for every φ we have that $\mathbb{N}^+ \setminus \text{Spec}(\varphi)$ is a spectrum. A first guess would be to check whether $\text{Spec}(\neg\varphi) = \mathbb{N}^+ \setminus \text{Spec}(\varphi)$. However, this is not true in general.

Let φ be a conjunction of the universal closures of axioms of fields. Then $\text{Spec}(\varphi) = \{p^m; p \text{ a prime and } m \in \mathbb{N}^+\}$. However,

$$\text{Spec}(\neg\varphi) = \mathbb{N}^+ \supsetneq \mathbb{N}^+ \setminus \text{Spec}(\varphi). \tag{2.2.1}$$

This is because for each cardinality we can pick an interpretation such that the resulting structure would not be a finite field.

In general $\mathrm{Spec}(\neg\varphi) \supseteq \mathbb{N}^+ \setminus \mathrm{Spec}(\varphi)$ since for every structure $\mathcal{A}$ of size $m \notin \mathrm{Spec}(\varphi)$ we have that $\mathcal{A} \not\models \varphi$ therefore $\mathcal{A} \models \neg\varphi$.

Whether spectra are closed under complementation is actually an open question. For this thesis it is actually **the open question**. It is called the **Asser's Spectrum problem**.

Conjecture 2.2.2 (Asser's spectrum problem)
Spectra are not closed under complementation.

If this conjecture were to be proven it would have vast consequences in complexity theory. From Corollary 3.3.2 which states that SPEC = **NE** it follows that if the conjecture was proved to be true we would also get **NE** $\neq$ **coNE**. Since deterministic complexity classes are closed under complementation, we get **E** $\neq$ **NE** and it is well-known this implies that **P** $\neq$ **NP**.

One way to see that **E** $\neq$ **NE** $\Rightarrow$ **P** $\neq$ **NP** is to consider $\mathbf{NP}_1$ the class of all **NP** languages that contain only words over the unary alphabet $\{1\}$. The only information each word in such a language gives is the number of ones in it. We will call these tally languages. Given a binary word w, we denote by 1^w the unary word whose length is the number represented by w. Note that if $n = |w|$, the length of 1^w is proportional to 2^n. Hence algorithms running in $2^{\mathcal{O}(n)}$-time on binary words correspond to algorithms that run in polynomial-time on the unary representations of words 1^w. Therefore $L \in \mathbf{NE} \Leftrightarrow L_1 \in \mathbf{NP}$ and analogously $L \in \mathbf{E} \Leftrightarrow L_1 \in \mathbf{P}$, where L_1 is the tally representation of the language L.

Now if the conjecture were to be proven and as a corollary we had some language $L \in \mathbf{NE} \setminus \mathbf{E}$ then we would have $L_1 \in \mathbf{NP} \setminus \mathbf{P}$ which would mean that **P** $\neq$ **NP** which is a fundamental problem in mathematics.

To summarize: if we expect the solution of Asser's spectrum problem to be negative then the problem is at least as hard as proving **P** $\neq$ **NP**.

2.2.2 Arithmetical operations on spectra

Since spectra are subsets of natural numbers it is also possible to consider adding and multiplying spectra element-wise.

Lemma 2.2.3
Let $A, B \in$ SPEC, then

$$A + B = \{a + b; a \in A, b \in B\} \in \mathrm{SPEC} \tag{2.2.2}$$

$$A \cdot B = \{a \cdot b; a \in A, b \in B\} \in \mathrm{SPEC}. \tag{2.2.3}$$

Proof. We assume that $\mathrm{Spec}(\varphi_A) = A$ and $\mathrm{Spec}(\varphi_B) = B$ and that τ_A is a vocabulary of φ_A and τ_B of φ_B.

For the addition we consider the vocabulary $\tau_+ = \{X, Y\} \cup \tau_A \cup \tau_B$ where X and Y are unary relational symbols. Now we define

$$\varphi_{A+B} = (\forall x)(X(x) \not\equiv Y(x)) \wedge \varphi_A^X \wedge \varphi_B^Y, \tag{2.2.4}$$

where the notation ψ^Z for a sentence ψ and a unary relational symbol Z is the relativization of ψ to Z and is defined by replacing every quantifier $(\forall v)$ by

$(\forall v \in Z)$ and $(\exists v)$ by $(\exists v \in Z)$. The models of φ_{A+B} are precisely the disjoint unions of pairs of a model of φ_A and a model of φ_B. Therefore $\mathrm{Spec}(\varphi_{A+B}) = A + B$.

Now for the multiplication we consider the vocabulary $\tau_\cdot = \{B, X, Y\} \cup \tau_A \cup \tau_B$ where X and Y are as before and B is a ternary relational symbol and define

$$\begin{aligned}\varphi_{A\cdot B} = {} & (\forall x \in X)(x \notin Y) \wedge (\forall y \in Y)(y \notin X) \wedge \tau_A^X \wedge \tau_B^Y \\ & \wedge (\forall x \in X)(\forall y \in Y)(\exists! z)(B(x, y, z)) \\ & \wedge (\forall z)(\exists! x \in X)(\exists! y \in Y)(B(x, y, z))).\end{aligned} \tag{2.2.5}$$

The finite models of $\varphi_{A\cdot B}$ are precisely those finite $\tau_\cdot$-structures $\mathcal{A}$ that contain the disjoint union of a model of φ_A with universe $X^{\mathcal{A}}$ and a model of φ_B with universe $Y^{\mathcal{A}}$ such that the whole universe of $\mathcal{A}$ is in bijection with $X^{\mathcal{A}} \times Y^{\mathcal{A}}$ and $B^{\mathcal{A}}$ is a graph of the bijection. It follows that $\mathrm{Spec}(\varphi_{A\cdot B}) = A \cdot B$. □

This lemma can be used multiple times to combine more than two spectra. The following corollary can be proved in a simmilar manner.

Corollary 2.2.4
Let $p \in \mathbb{N}^+[x_1, \ldots, x_n]$ be a polynomial and $X_1, \ldots, X_n \in \textsc{Spec}$. Then

$$p(X_1, \ldots, X_n) = \{p(a_1, \ldots, a_n); a_i \in X_i\} \in \textsc{Spec}. \tag{2.2.6}$$

Now we know that spectra are closed under union, intersection, element-wise addition and multiplication and also under every polynomial mapping. From this and the fact that (co)finite sets of natural numbers are spectra we get many examples of spectra. Numbers congruent to j mod k, squares, cubes, numbers that are both squares and cubes, fourth-powers and so on.

What other operations on spectra result in spectra? It turns out that spectra are closed under every non-decreasing exponential-time function. See Corollary 3.4.2 for more details.

In the next chapter we define a new notion of Σ_1^1-definable functions which generalizes all of these classes of operations. We prove that Σ_1^1-definable function are closed under composition (with some mild assumptions) and under a form of iteration as in Cobham's theorem. We use this to prove Fagin's theorem and hence also the Jones-Selman characterization of spectra as **NE** sets.

3. Σ_1^1-definable functions

To study iterations of operations on spectra it appears to be more natural to aim first at characterizing generalized spectra. In this chapter we define Σ_1^1-definable functions. We will use Cobham's theorem 1.2.2 to show that all functions in **FP** are Σ_1^1-definable and we will then use it to give a new proof of Fagin's theorem. We later show in what sense are Σ_1^1-definable functions a generalization of the earlier examples of operations on SPEC.

3.1 Basic notions

Definition 3.1.1 (Coding binary words by relations)**.** Let $(A, <)$ be a finite linearly ordered set, let $<^n$ be the lexicographical ordering on A^n and let $R \subseteq A^n$ be nonempty. We define the binary word coded by the relation R as $*R \in \{0,1\}^*$ such that

$$|*R| = \mathrm{ord}(\max_{<^n} R) \tag{3.1.1}$$

$$*R_i = \chi_R(\mathrm{ord}^{-1}(i)), \text{ for } i \in \{0, \ldots, |*R| - 1\}, \tag{3.1.2}$$

where χ_R is the characteristic function of R and $\mathrm{ord} : A^n \cong \{0, \ldots, |*R| - 1\}$ is an order preserving bijection and $*R_i$ is the i-th bit of $*R$.

Definition 3.1.2 (Σ_1^1-definable function)**.** We say $f : (\{0,1\}^*)^k \to \{0,1\}^*$ is Σ_1^1-definable iff there exist Σ_1^1-sentence θ_f in the vocabulary $\tau := \{U_1, \ldots, U_k, V, <\}$, where $U_1, \ldots, U_k, V$ are unary relational symbols, $<$ is a binary relational symbol and for every τ-structure $\mathcal{A}$ with nonempty interpretations of τ, where $<^{\mathcal{A}}$ is an linear ordering, we have

$$\mathcal{A} \models \theta_f(U_1, \ldots, U_k, V, <) \tag{3.1.3}$$

iff

$$f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}) = *V^{\mathcal{A}}. \tag{3.1.4}$$

We call θ_f **a defining Σ_1^1-sentence of the function** f. We denote the set of all Σ_1^1-definable functions $\mathbf{F}\Sigma_1^1$.

Many functions in mathematics are defined by recursion. E.g. multiplication as repeated addition and exponentiation as repeated multiplication. We want to be able to imitate similar recursions while constructing Σ_1^1-defining formulas. The following lemma helps us define functions by defining each of their intermediate values in such a recursive scheme.

Lemma 3.1.3 (Indexed relational symbols are "substitutable")
Let $\theta(\bar{x})$ be an $\{S_1, \ldots, S_k, R\}$-formula, where R is c-ary relational symbol. Then there exists $\{S_1, \ldots, S_k, R_-\}$-formula $\psi(\bar{x}, y)$, where R_- is $(c+1)$-ary such that for each $\{S_1, \ldots, S_k, R_-\}$-structure $\mathcal{A}$ and $\bar{a}, i \in A$

$$\mathcal{A} \models \psi(\bar{a}, i) \Leftrightarrow (A, S_1^{\mathcal{A}}, \ldots, S_k^{\mathcal{A}}, R_i^{\mathcal{A}}) \models \theta(\bar{a}), \tag{3.1.5}$$

where $R_i^{\mathcal{A}} := \{(x_2, \ldots, x_{c+1}); (i, x_2, \ldots, x_{c+1}) \in R_-^{\mathcal{A}}\}$.

Proof. (sketch) We can define a translation $(-)'$ solely on atomic formulas as

$$(-)' : \varphi(\bar{z}) \mapsto \begin{cases} R_{-}(y, \bar{z}) & \varphi = R(\bar{z}) \\ \varphi(\bar{z}) & \text{otherwise} \end{cases} \tag{3.1.6}$$

and let it commute with all logical connectives and quantifiers. It is not hard to see that by putting $\psi := \theta'$ we have (3.1.5). □

Lemma 3.1.4
The following functions are Σ_1^1-definable:

(i) $(x, y) \mapsto x + y$

(ii) $(x, y) \mapsto x \cdot y$

(iii) $x \mapsto \lfloor \frac{x}{2} \rfloor$

(iv) $x \mapsto \lfloor \sqrt{x} \rfloor$

(v) $(x, y) \mapsto \lfloor \frac{x}{y} \rfloor$

(vi) $\exp : x \mapsto 2^x$.

Proof. (sketch) We shall only outline the proof since the details are easy to fill. We can also assume that the underlying set of each model is just a finite initial segment of $\mathbb{N}^+$ with its natural ordering.

The function (i) can be Σ_1^1-defined in an analogous way to how boolean circuit for an adder is implemented. That is, for each two bits we compute their addition that is their XOR and their carry and use that carry for the addition of the next two bits. The carry can be "stored" in another unary relation with symbol C which we can existentially quantify.

For (ii) we can formalize the grade-school multiplication algorithm. We use the Σ_1^1-formula for (i) to compute the intermediate results. For "storing" them we can use Lemma 3.1.3 and existentially quantify a binary relation with symbol M_{-} which would be for each element a understood as the a-th unary relation denoted M_a. Then we can set the output relation with symbol V to be equal to the last of M_a's.

The (iii) - (v) can be Σ_1^1-defined analogously using the Σ_1^1-defining formulas for (i) and (ii).

To Σ_1^1-define (vi) we can use the same technique as in (ii) only on the Σ_1^1-formula from (ii) itself by existentially quantifying a binary relation E_{-} treated as an indexed tuple of unary relations, such that E_a would be understood as a unary relation symbol coding 2^a. We can then set V to be interpreted the same as E_x, where U codes the binary code of number x. Singling out the element x coded by U can be also done iteratively. □

3.2 Σ_1^1-definability of FP

Note that the last item in Lemma 3.1.4 implies that $\mathbf{F}\Sigma_1^1$ is not contained in **FP**. However, in this section we prove that $\mathbf{F}\Sigma_1^1$ contains **FP**. We will do so

in a series of lemmas and theorems. We show that a subset of $\mathbf{F}\Sigma_1^1$ satisfies the conditions from Theorem 1.2.2 by finding a bit more general conditions for composing Σ_1^1-definable functions and constructing Σ_1^1-definable functions by limited recusion on notation.

We start by showing that the generating functions of **FP** are Σ_1^1-definable.

Lemma 3.2.1
The following functions are Σ_1^1-definable:

1. $s_i : x \mapsto xi$, where $i \in \{0, 1\}$
2. $\pi_i^k : (x_1, \ldots, x_k) \mapsto (x_i)$, where $i \in \{1, \ldots, k\}$
3. $\# : (x, y) \mapsto 2^{|x| \cdot |y|}$

Proof. For s_i we put the Σ_1^1-defining sentence to be

$$\begin{aligned}\theta_{1,i} = (\exists_2 S)(\forall x)(\forall y)(S(x,y) \leftrightarrow (x < y \wedge (\forall z < y)(z \leq x))) \\ \wedge (\forall x)(x \in V \leftrightarrow (\\ (x \in U \wedge x < \max U) \\ \vee (x = \max U \wedge x =_i x) \\ \vee (\forall y)(S(\max U, y) \rightarrow x = y)),\end{aligned} \tag{3.2.1}$$

where $=_i$ denotes $=$ for $i = 1$ and $\neq$ for $i = 0$. The sentence formalizes that $*V$ is one bit longer that $*U$ and that the rightmost bit is i.

π_i^k can be Σ_1^1-defined by the formula $\theta_{2,i}^k = (\forall x)(V(x) \leftrightarrow U_i(x))$.

The smash function $\#$ can be Σ_1^1-defined by the sentence

$$\begin{aligned}\theta_3 := (\exists_3 B)((\forall u_1 < \max U_1)(\forall u_2 < \max U_2)(\exists! v < \max V)(B(u_1, u_2, v)) \\ \wedge (\forall v < \max V)(\exists! u_1 \ < \max U_1)(\exists! u_2 < \max U_2)(B(u_1, u_2, v)) \\ \wedge (\forall v \in V)((v < \max V) \rightarrow (\forall w)(v \leq w))).\end{aligned} \tag{3.2.2}$$

The first two lines just state that B is the graph of a bijective function from the cartesian product of bits of $*U_1$ and $*U_2$ to the bits of $*V$ and the last line states that the first bit of $*V$ is the only one with value 1, therefore $*V$ is of the form $10 \ldots 00$. □

We would now like to show that Σ_1^1-definable functions are closed under composition and limited recursion on notation. We would like to prove it in a straightforward way by essentially composing their graphs and using the conjunction of the given Σ_1^1-defining formulas. However the intermediate values can be longer than the model can encode by a unary relation. In limited recursion on notation the intermediate values are bounded in length by a polynomial in the length of inputs and we can use relations of higher arities to encode them. Now follows a lemma of a technical nature which solves this issue by allowing us to use Σ_1^1-defining formulas for relational symbols with higher arities. Interpretation of these can then code polynomially long words.

Lemma 3.2.2 (Inflating arities)
Let θ_f Σ_1^1-define the function $f(x_1, \ldots, x_k)$ and $c \in \mathbb{N}^+$. Let $\tilde{\tau} := \{\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{V}, <\}$ such that relational symbols of $\tilde{\tau}$ are all c-ary except $<$ which

is binary. Then there exists a Σ^1_1-sentence $\theta_f^{(c)}$ in the vocabulary $\tilde{\tau}$ and for each $\tilde{\tau}$-structure $\mathcal{A}$ linearly ordered by $<^{\mathcal{A}}$:

$$\mathcal{A} \models \theta_f^{(c)}(\tilde{U}_1, \dots, \tilde{U}_k, \tilde{V}) \Leftrightarrow f(*\tilde{U}_1^{\mathcal{A}}, \dots, *\tilde{U}_k^{\mathcal{A}}) = *\tilde{V}^{\mathcal{A}}. \tag{3.2.3}$$

Proof. We define the translation $(-)^{(c)}$ on Σ^1_1-formulas in the vocabulary $\tau := \{U_1, \dots, U_k, V, <\}$ by induction on the complexity of the formula. We will proceed by replacing variables of these formulas with c-tuples of new variables. First for atomic formulas

- $(x = y)^{(c)} = \bigwedge_{i=1}^{c} x_i = y_i$, where x_i's and y_i's are newly introduced variables,
- $(x < y)^{(c)} = \bigvee_{i=1}^{c}(\bigwedge_{j=1}^{i-1}(x_j = y_j) \wedge x_i < y_i)$, where x_i's and y_i's are newly introduced variables,
- $(R(x_1, \dots, x_a))^{(c)} = \tilde{R}(x_{1,1}, \dots, x_{1,c}; \dots; x_{a,1}, \dots, x_{a,c})$, where R is an a-ary relational symbol or a second-order variable, $\tilde{R}$ is $c \cdot a$-ary and $x_{i,j}$'s are newly introduced variables.

The newly introduced variables are equal iff they have the same index and they substitute the same original variable. The semicolons in the last item are just for better readability.

Now assume that the translation for θ_1 and θ_2 is defined, then

- $(\neg\theta_1)^{(c)} = \neg\theta_1^{(c)}$
- $(\theta_1 \diamond \theta_2)^{(c)} = \theta_1^{(c)} \diamond \theta_2^{(c)}$, where $\diamond \in \{\wedge, \vee\}$,
- $((Qx)\theta_1(x, y_1, \dots, y_a))^{(c)} = (Qx_1)\dots(Qx_c)\theta_1^{(c)}(x_1, \dots, x_c; y_{1,1}, \dots, y_{1,c}; \dots)$, where $Q \in \{\exists, \forall\}$,
- $((\exists_a R)\theta_1(R, S_1, \dots, S_l, \dots))^{(c)} = (\exists_{c\cdot a}\tilde{R})\theta_1^{(c)}(\tilde{R}, \tilde{S}_1, \dots, \tilde{S}_l, \dots)$.

Now for each $\tilde{\tau}$-structure $\mathcal{A}$ we define $\mathcal{B} := (A^c, \tilde{U}_1^{\mathcal{A}}, \dots, \tilde{U}_k^{\mathcal{A}}, \tilde{V}^{\mathcal{A}}, <')$ interpreting the vocabulary τ, where $<'$ is the lexicographical order on c-tuples constructed using $<^{\mathcal{A}}$.

Claim: Let $\varphi(x_1, \dots, x_a)$ be a Σ^1_1-formula in the vocabulary τ, let $\mathcal{A}$ be a $\tilde{\tau}$-structure and $v_{1,1}, \dots, v_{1,c}; \dots; v_{a,1}, \dots, v_{a,c} \in A$. Then

$$\mathcal{A} \models \varphi^{(c)}(v_{1,1}, \dots, v_{1,c}; \dots; v_{a,1}, \dots, v_{a,c}) \tag{3.2.4}$$

$$\Updownarrow$$

$$\mathcal{B} \models \varphi((v_{1,1}, \dots, v_{1,c}), \dots, (v_{a,1}, \dots, v_{a,c})). \tag{3.2.5}$$

Specifically for φ a Σ^1_1-sentence we simply have

$$\mathcal{A} \models \varphi^{(c)} \Leftrightarrow \mathcal{B} \models \varphi. \tag{3.2.6}$$

This claim can be straightforwardly proved by induction on the formula φ (which could have some free second-order variables whose arities are inflated as well).

Let $R^{\mathcal{A}}$ be some interpretation of $(i \cdot c)$-ary relational symbol R in $\mathcal{A}$ and let $\tilde{R}^{\mathcal{B}}$ be the corresponding interpretation in $\mathcal{B}$. Since $*R^{\mathcal{A}}$ is defined using characteristic function of the corresponding lexicographical ordering on $(i \cdot c)$-tuples and $*\tilde{R}^{\mathcal{B}}$ is defined using the corresponding lexicographical ordering on i-tuples of c-tuples it easily follows that $*R^{\mathcal{A}} = *\tilde{R}^{\mathcal{B}}$.

Now since θ_f is a sentence in the vocabulary $\{U_1, \ldots, U_k, V, <\}$, then $\mathcal{A} \models \theta_f^{(c)}(\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{V}) \Leftrightarrow \mathcal{B} \models \theta_f(U_1, \ldots, U_k, V) \Leftrightarrow f(*U_1^{\mathcal{B}}, \ldots, *U_k^{\mathcal{B}}) = *V^{\mathcal{B}} \Leftrightarrow f(*\tilde{U}_1^{\mathcal{A}}, \ldots, *\tilde{U}_k^{\mathcal{A}}) = *\tilde{V}^{\mathcal{A}}$ which proves the lemma. □

The following definition will come in handy when we use Lemma 3.2.2 to construct new Σ_1^1-defining formulas. When we "inflate" the arities of one formula then we cannot directly plug in the input relations into it, we need c-ary versions of these relations. These are defined so that they code the same binary word.

Definition 3.2.3 (extd_c)**.** Let $c \in \mathbb{N}^+$, then the $\{X, Y\}$-formula for X unary and Y c-ary relational symbols $\mathrm{extd}_c(X, Y)$ is defined as

$$(\forall x_1) \ldots (\forall x_c)((x_c \in X \wedge (\forall z)(\bigwedge_{i=1}^{c-1} x_i \leq z)) \leftrightarrow (x_1, \ldots, x_c) \in Y). \tag{3.2.7}$$

It is not hard to show, that for any $\mathcal{A} \models \mathrm{extd}_c(X, Y)$ we have $*X^{\mathcal{A}} = *Y^{\mathcal{A}}$.

Theorem 3.2.4 (Composition of Σ_1^1-definable functions)
Let $f : (\{0,1\}^*)^k \to \{0,1\}$ and $g : (\{0,1\}^*)^{k+1} \to \{0,1\}^*$ be Σ_1^1-definable and let there exist $c \in \mathbb{N}^+$ such that for all $x_1, \ldots, x_k \in \{0,1\}^*$

$$|f(x_1, \ldots, x_n)| \leq \max(|x_1|, \ldots, |x_k|, |g(x_1, \ldots, x_k, f(x_1, \ldots, x_k))|, 2)^c \tag{3.2.8}$$

then $g(x_1, \ldots, x_k, f(x_1, \ldots, x_k))$ is Σ_1^1-definable.

Proof. Let θ_f, θ_g be the Σ_1^1-defining formulas of f and g respectively in the vocabularies $\tau = \{U_1, \ldots, U_k, V\}$ and $\tau \cup U_{k+1}$. We then define

$$\begin{aligned} \theta_0 :=& (\exists_c V_f)(\exists_c \tilde{U}_1) \ldots (\exists_c \tilde{U}_k)(\exists_c \tilde{V}) \, (\bigwedge_{i=1}^{k} \mathrm{extd}_c(U_i, \tilde{U}_i) \wedge \mathrm{extd}_c(V, \tilde{V}) \\ & \wedge \theta_f^{(c)}(\tilde{U}_1, \ldots, \tilde{U}_k, V_f) \wedge \theta_g^{(c)}(\tilde{U}_1, \ldots, \tilde{U}_k, V_f, \tilde{V})). \end{aligned} \tag{3.2.9}$$

If $g(\epsilon, \epsilon, \ldots, f(\epsilon, \ldots, \epsilon)) = \epsilon$, where ϵ denotes the empty word, we put

$$\theta := \theta_0 \vee ((\exists x)(\forall y)(x = y) \wedge (\forall x) x \in V), \tag{3.2.10}$$

otherwise we put $\theta := \theta_0$.

We will proceed to prove that θ Σ_1^1-defines $g(x_1, \ldots, x_k, f(x_1, \ldots, x_k))$. We just need

$$g(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})) = *V^{\mathcal{A}} \Leftrightarrow \mathcal{A} \models \theta \tag{3.2.11}$$

for $|A| \geq 2$, since the case for $|A| = 1$ is covered by (3.2.10).

"⇒" We assume $g(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})) = *V^{\mathcal{A}}$. We will construct the witnessing expansion of $\mathcal{A}$ denoted $\mathcal{A}^+$.

Since we have

$$\begin{aligned} |f(*U_1{}^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})| &\leq \max(|*U_1^{\mathcal{A}}|, \ldots, |*U_k^{\mathcal{A}}|, |*V^{\mathcal{A}}|, 2)^c \\ &\leq (|A| - 1)^c \\ &\leq |A|^c - 1 \end{aligned} \tag{3.2.12}$$

we can put $V_f^{\mathcal{A}^+}$ such that $*V_f^{\mathcal{A}^+} = f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})$. There also exist some witnessing interpretations of the second-order variables quantified in θ_f and θ_g because they are Σ_1^1-defining. Finally for each $\tilde{R} \in \{\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{V}\}$ we put $\tilde{R}^{\mathcal{A}^+}$ such that $\mathcal{A}^+ \models \mathrm{extd}_c(R, \tilde{R})$.

Let the $\{\tilde{U}_1, \ldots, \tilde{U}_k, V_f, <\}$-reduct of $\mathcal{A}^+$ be denoted $\mathcal{A}'$. Since θ_f Σ_1^1-defines f and $f(*\tilde{U}_1^{\mathcal{A}'}, \ldots, *\tilde{U}_k{}^{\mathcal{A}'}) = *\tilde{V}_f^{\mathcal{A}'}$, then by Lemma 3.2.2 we have $\mathcal{A}' \models \theta_f^{(c)}$. Therefore $\mathcal{A}^+ \models \theta_f^{(c)\mathrm{FO}}$ which in turn yields $\mathcal{A} \models \theta_f^{(c)}$.

Analogously taking the $\{\tilde{U}_1, \ldots, \tilde{U}_k, V_f, \tilde{V}, <\}$-reduct of $\mathcal{A}^+$ denoted $\mathcal{A}''$ we have $\mathcal{A}'' \models \theta_g^{(c)}$ and so $\mathcal{A} \models \theta_g^{(c)}$. This is enough to conclude $\mathcal{A} \models \theta$.

"$\Leftarrow$" On the other hand assume $\mathcal{A} \models \theta$. We will call the witnessing expansion $\mathcal{A}^+$ and the reducts corresponding to the ones from the "$\Rightarrow$" part $\mathcal{A}'$ and $\mathcal{A}''$.

We then have

$$\mathcal{A} \models \theta \Rightarrow \mathcal{A}^+ \models \theta^{FO} \tag{3.2.13}$$

$$\Rightarrow \mathcal{A}' \models \theta_f^{(c)}(\tilde{U}_1, \ldots, \tilde{U}_k, V_f) \tag{3.2.14}$$

$$\overset{\text{Lemma 3.2.2}}{\Rightarrow} f(*\tilde{U}_1^{\mathcal{A}}, \ldots, *\tilde{U}_k^{\mathcal{A}}) = *V_f^{\mathcal{A}} \tag{3.2.15}$$

and

$$\mathcal{A} \models \theta \Rightarrow \mathcal{A}^+ \models \theta^{FO} \tag{3.2.16}$$

$$\Rightarrow \mathcal{A}' \models \theta_g^{(c)}(\tilde{U}_1, \ldots, \tilde{U}_k, V_f, V) \tag{3.2.17}$$

$$\overset{\text{Lemma 3.2.2}}{\Rightarrow} g(*\tilde{U}_1^{\mathcal{A}}, \ldots, *\tilde{U}_k^{\mathcal{A}}, *V_f^{\mathcal{A}}) = *\tilde{V}^{\mathcal{A}}. \tag{3.2.18}$$

It follows from (3.2.15) and (3.2.18) that

$$g(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})) = *V^{\mathcal{A}}, \tag{3.2.19}$$

which proves the theorem. □

Theorem 3.2.5 (Limited recursion on notation of Σ_1^1-definable functions.) Assume $h_0(x_1, \ldots, x_k, x, y)$, $h_1(x_1, \ldots, x_k, x, y)$, $g(x_1, \ldots, x_n)$ are Σ_1^1-definable functions, defined by formulas $\theta_{h_0}, \theta_{h_1}, \theta_g$ respectively. Let $c \in \mathbb{N}^+$. Assume the function $f : (\{0,1\}^*)^{k+1} \to \{0,1\}^*$ satisfies

$$f(x_1, \ldots, x_k, \epsilon) := g(x_1, \ldots, x_k) \tag{3.2.20}$$

$$f(x_1, \ldots, x_k, s_0(x)) := h_0(x_1, \ldots, x_k, x, f(x_1, \ldots, x_k, x)) \tag{3.2.21}$$

$$f(x_1, \ldots, x_k, s_1(x)) := h_1(x_1, \ldots, x_k, x, f(x_1, \ldots, x_k, x)), \tag{3.2.22}$$

and for all $x_1, \ldots, x_k, x \in \{0,1\}^*$ we have

$$|f(x_1, \ldots, x_k, x)| \leq \max(|x_1|, \ldots, |x_k|, |x|, 2)^c. \tag{3.2.23}$$

Then f is Σ_1^1-definable.

Proof. We want to find some Σ^1_1-sentence θ which would define the process from limited recursion on notation and prove it to be a Σ^1_1-defining sentence of f.

To compute the value $f(x_1, \ldots, x_k, x)$ we start by computing $y_0 = g(x_1, \ldots, x_k)$ then $y_1 = h_i(x_1, \ldots, x_k, \epsilon, y_0)$, where i is the first bit, and so on for $|x|$ many steps. The existentially quantified relations will encode this process, in particular the values $y_0, \ldots, y_{|x|}$ are encoded by the interpretations of $\tilde{V}_i$'s.

To formalize the process we use the binary symbol S, which will represent the successor relation, and the unary symbol M_{in}, which will represent the singleton containing the minimum.

We define

$$\begin{aligned}
\theta_0 :=&(\exists_{1+c}\tilde{V}_-)(\exists_2 W_-)(\exists_{1+c}\tilde{W}_-)(\exists_1 M_{in})(\exists_2 S)(\exists_2 C)(\\
&(\forall x)(x \in M_{in} \leftrightarrow (\forall y)(x \leq y)) \\
&\wedge (\forall x)(\forall y)(S(x,y) \leftrightarrow (x < y \wedge (\forall z)(x < z \rightarrow y \leq z))) \\
&\wedge (\forall i)(\forall x)(x \in W_i \leftrightarrow ((x \in U \wedge x < i) \vee (x = i))) \\
&\wedge (\forall i)\mathrm{extd}'_c(W_i, \tilde{W}_i) \\
&\wedge \mathrm{extd}'_c(V, \tilde{V}_{\max U}) \\
&\wedge (\forall x)(\forall y)((S(x,y) \vee y \in M_{in}) \rightarrow (\\
&\quad (y \in M_{in} \wedge \theta'^{(c)}_g(\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{V}_y)) \\
&\quad \vee (y \notin M_{in} \wedge y \leq \max U \wedge x \notin U \wedge \theta'^{(c)}_{h_0}(\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{W}_x, \tilde{V}_x, \tilde{V}_y)) \\
&\quad \vee (y \notin M_{in} \wedge y \leq \max U \wedge x \in U \wedge \theta'^{(c)}_{h_1}(\tilde{U}_1, \ldots, \tilde{U}_k, \tilde{W}_x, \tilde{V}_x, \tilde{V}_y)) \\
&\quad \vee (y \notin M_{in} \wedge y > \max U) \\
&))),
\end{aligned} \tag{3.2.24}$$

where $\theta'_g, \theta'_{h_0}, \theta'_{h_1}, \mathrm{extd}'_c$ are obtained by applying the Lemma (3.1.3) enough times to make all indexed relations substitutable. If $f(\epsilon, \ldots, \epsilon) = \epsilon$ we then put

$$\theta := \theta_0 \vee ((\exists x)(\forall y)x = y \wedge (\forall x)x \in V) \tag{3.2.25}$$

otherwise we put $\theta := \theta_0$.

To prove that θ Σ^1_1-defines f we first need to know that the values used in the recursion are encodable by a c-ary relation. Let $\mathcal{A}$ be an $\{U_1, \ldots, U_k, V, <\}$-structure linearly ordered by $<^{\mathcal{A}}$. We then have

$$\begin{aligned}
|g(*U_1^{\mathcal{A}}, \ldots, *U_k{}^{\mathcal{A}})| &= |f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, \epsilon)| \\
&\leq \max(|*U_1^{\mathcal{A}}|, \ldots, |*U_k^{\mathcal{A}}|, 2)^c \\
&\leq (|A| - 1)^c \\
&\leq |A|^c - 1
\end{aligned} \tag{3.2.26}$$

and for every $x \in \{0,1\}^*$ such that $s_i(x)$ is encodable by a c-ary relation we have

$$\begin{aligned}
|h_i(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, x, f(x))| &= |f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, s_i(x))| \\
&\leq \max(|*U_1^{\mathcal{A}}|, \ldots, |*U_k^{\mathcal{A}}|, |s_i(x)|, 2)^c \\
&\leq (|A| - 1)^c \\
&\leq |A|^c - 1.
\end{aligned} \tag{3.2.27}$$

To prove that θ Σ_1^1-defines $f(x_1, \ldots, x_k, x)$ we just need

$$f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, *U^{\mathcal{A}}) = *V^{\mathcal{A}} \Leftrightarrow \mathcal{A} \models \theta(U_1^{\mathcal{A}}, \ldots, U_k^{\mathcal{A}}, V^{\mathcal{A}}) \tag{3.2.28}$$

for $|A| \geq 2$ since the case for $|A| = 1$ is covered by (3.2.25).

"$\Rightarrow$" We assume $f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, *U^{\mathcal{A}}) = *V^{\mathcal{A}}$. We now need to construct the witnessing expansion $\mathcal{A}^+$.

The witnessing interpretations of explicitly quantified relational symbols other than $\tilde{V}_-$ are uniquely determined from $U_1^{\mathcal{A}}, \ldots, U_k^{\mathcal{A}}, V^{\mathcal{A}}, <^{\mathcal{A}}$. Moreover the witnessing interpretations of relational symbols quantified in $\theta_g'^{(c)}, \theta_{h_i}'^{(c)}, \mathrm{extd}_c'$ exist because of the construction of those sentences. So we just need to find the witnessing interpretation of $\tilde{V}_-$.

Without loss of generality we have $A = \{0, \ldots, n-1\}$ ordered exactly as written. Thanks to the bound (3.2.26) we can put $\tilde{V}_0^{\mathcal{A}^+}$ such that

$$*\tilde{V}_0^{\mathcal{A}^+} = g(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}). \tag{3.2.29}$$

Simmilary because of (3.2.27) we can put $V_j^{\mathcal{A}^+}$, where $j \in \{1, \ldots, |*U^{\mathcal{A}}|\}$ such that

$$*\tilde{V}_j^{\mathcal{A}^+} = h(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, *W_{j-1}^{\mathcal{A}^+}, *V_{j-1}^{\mathcal{A}^+}). \tag{3.2.30}$$

From the way θ_g', θ_h' are constructed and Lemma 3.2.2 we have $\mathcal{A}^+ \models \theta^{FO}$. Therefore $\mathcal{A} \models \theta$.

"$\Leftarrow$" We assume that $\mathcal{A} \models \theta$ and denote $\mathcal{A}^+$ the witnessing expansion. Again without loss of generality we have $A = \{0, \ldots, n-1\}$ ordered exactly as written. Since θ_g Σ_1^1-defines g then by Lemma 3.1.3 we have $*\tilde{V}_0^{\mathcal{A}^+} = g(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}})$. Similarly θ_{h_i} defines $h_i, i \in \{0, 1\}$, therefore by Lemma 3.1.3 we have for $j \in \{1, \ldots, |*U^{\mathcal{A}}|\}$: $*\tilde{V}_j^{\mathcal{A}^+} = h_{b_j}(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, *\tilde{W}_{j-1}^{\mathcal{A}^+}, *\tilde{V}_{j-1}^{\mathcal{A}^+})$, where b_j is the j-th bit of $*U^{\mathcal{A}}$.

Since $*\tilde{W}_j^{\mathcal{A}^+}$ is defined to be the first j bits of $*U^{\mathcal{A}}$ and $*V^{\mathcal{A}} = *\tilde{V}_{|*U^{\mathcal{A}}|}^{\mathcal{A}^+}$ we have

$$*V^{\mathcal{A}} = f(*U_1^{\mathcal{A}}, \ldots, *U_k^{\mathcal{A}}, *U^{\mathcal{A}}). \tag{3.2.31}$$

Which proves the theorem. □

Now we can easily show that every function in **FP** is Σ_1^1-definable.

Theorem 3.2.6

$$\mathbf{FP} \subsetneq \mathbf{F}\Sigma_1^1 \tag{3.2.32}$$

Proof. The inequalities needed in Theorems 3.2.4 and 3.2.5 are trivially met for any function in **FP** since for any $f(x_1, \ldots, x_k) \in \mathbf{FP}$ we have a polynomial p such that $|f(x_1, \ldots, x_k)| \leq p(|x_1|, \ldots, |x_k|)$. This is because otherwise the Turing machine computing the function would have only polynomial-time to write the whole superpolynomially long output and that is impossible.

In Lemma 3.2.1 and Theorems 3.2.4 and 3.2.5 we verified that $\mathbf{FP} \cap \mathbf{F}\Sigma_1^1$ satisfies the assumptions of the Cobham's characterization of **FP** (Theorem 1.2.2) hence $\mathbf{FP} \subseteq \mathbf{F}\Sigma_1^1$.

Moreover $\exp \in \mathbf{F}\Sigma_1^1$ but $\exp \notin \mathbf{FP}$, so the inclusion is strict. □

3.3 Fagin's theorem

Now everything is ready and we can proceed to prove Fagin's theorem. Let us note that Fagin's theorem as usually stated considers generalized spectra only over a non-empty vocabulary. This is important only because the usual encoding of finite structures encodes models without relations as the binary number representing the size of the set, which is exponentially shorter then the codes of models over non-empty vocabulary. To make the statement of the theorem more clean we assume that sets without relations are encoded by a unary number representing their size. This allows us to include the empty vocabulary in the statement of the theorem.

For a class of binary structures $\mathcal{K}$ we will denote by $\mathcal{K}^*$ the set of binary codes of every linearly ordered τ-structure $(\mathcal{A}, <)$, where $\mathcal{A} \in \mathcal{K}$ and $<$ is a linear ordering on A. By a binary code of an linearly ordered τ-structure $(\mathcal{A}, <)$ we mean a binary word which starts with unary representation of the length of $\mathcal{A}$ delimited by 0 and followed by the characteristic table for each $R \in \tau$ in some fixed order. We will denote it $\mathrm{code}_\tau(\mathcal{A}, <)$.

The delimiting 0 is present only for structures with non-empty vocabulary. That means for an $\varnothing$-structure $\mathcal{A}$ we have $\mathrm{code}_\varnothing(\mathcal{A}, <) = \underbrace{1 \dots 1}_{\times |A|}$.

Theorem 3.3.1 (Fagin)
Let $\mathcal{K}$ be an isomorphism-closed class of τ-structures. Then

$$\mathcal{K}^* \in \mathbf{NP} \Leftrightarrow \mathcal{K} \in \textsc{GenSpec}. \tag{3.3.1}$$

Proof. "$\Leftarrow$" When $\mathcal{K} \in \textsc{GenSpec}$, it is not hard to find a nondeterministic algorithm deciding for any structure whether $\mathrm{code}_\tau(A, <) \in \mathcal{K}^*$ in polynomial-time. Since $\mathcal{A} \in \mathcal{K} \Leftrightarrow \mathcal{A} \models \varphi_\mathcal{K}$, where $\varphi_\mathcal{K}$ is the defining formula of the generalized spectrum $\mathcal{K}$ we can just guess the witnessing relations and then check whether the resulting expansion satisfies $\varphi^{\mathrm{FO}}_\mathcal{K}$.

To do so, we can just enumerate every c-tuple of elements of A, where c is the number of quantifiers of $\varphi^{\mathrm{FO}}_\mathcal{K}$ and check whether the quantifier-free part of $\varphi^{\mathrm{FO}}_\mathcal{K}$ is true with this evaluation of variables. The last part can be done in polynomial-time since it only requires to check a constant number of entries of $\mathrm{code}_\tau(\mathcal{A}, <)$. There are n^c c-tuples of A, so the final complexity of this algorithm will be $\mathcal{O}(n^c)$.

"$\Rightarrow$" We assume $\mathcal{K}^* \in \mathbf{NP}$. We want to prove $\mathcal{K}$ is a generalized spectrum. Since $\mathcal{K}^* \in \mathbf{NP}$ then there is $L' \in \mathbf{P}$ and $d \in \mathbb{N}^+$ such that

$$x \in \mathcal{K}^* \Leftrightarrow \exists w, |w| \leq |x|^d : (x, w) \in L'. \tag{3.3.2}$$

We have $\chi_{L'} \in \mathbf{FP}$, where $\chi_{L'}$ is the characteristic function of L' with two arguments for both the witness and potential member of $\mathcal{K}^*$. By Theorem 3.2.6 $\chi_{L'}$ is Σ^1_1-definable. Let $\theta'_L(X, W, V)$ be its Σ^1_1-defining formula.

We pick $c > d$ such that for every linearly ordered τ-structure $(\mathcal{A}, <)$ with at least two elements we have that $\mathrm{code}_\tau(A, <)$ is encodable by an c-ary relation on A. Notice that the code of a structure is Σ^1_1-definable. That is, there exists a Σ^1_1-formula $\mathrm{enc}_c(X)$ such that for an c-ary relational symbol X we have

$$(\mathcal{A}, X^\mathcal{A}, <) \models \mathrm{enc}_c(X) \Leftrightarrow *X^\mathcal{A} = \mathrm{code}_\tau(\mathcal{A}, <). \tag{3.3.3}$$

Such a formula can by constructed by formalizing there is an initial segment of $*X^{\mathcal{A}}$ containing only 1's delimited by 0 that has a bijection with the universe A and that the tables of other relations $\mathcal{A}$ follow. Now we claim that

$$\varphi_{\mathcal{K}} = (\exists_2 <)(\exists_c X)(\exists_c W)(\exists_c O)(\mathrm{enc}_c(X) \wedge \mathrm{one}_c(O) \wedge \mathrm{lin}(<) \wedge \theta^{(c)}_{L'}(X, W, O)) \vee \varphi_1 \quad (3.3.4)$$

is a Σ^1_1-defining formula of the generalized spectrum $\mathcal{K}$, where φ_1 is a disjunction of all sentences describing one-element structures whose codes are in $\mathcal{K}^*$ up to isomorphism, lo($<$) states that $<$ is a linear order and $\mathrm{one}_c(O)$ just states that $*O^{\mathcal{A}} = 1$.

From the construction of enc_c and $\theta_{L'}$ we have that it defines structures from $\mathcal{K}$ with at least two elements. One-element structures are defined additionally by φ_1. This proves the theorem. □

With Fagin's theorem in hand the next characterization of spectra follows easily. This corollary has been proved independently in [JS74].

Corollary 3.3.2 (Jones, Selman)

$$\textsc{Spec} = \mathbf{NE} \quad (3.3.5)$$

Proof. This follows from Theorem 3.3.1 and from the fact that

$$L \in \textsc{Spec} \Leftrightarrow \exists \mathcal{K} \in \textsc{GenSpec}_{\varnothing} : L_1 = \mathcal{K}^*, \quad (3.3.6)$$

where L_1 denotes the tally langauge corresponding to L and $\textsc{GenSpec}_{\varnothing}$ denotes the set of generalized spectra over the empty vocabulary. The "$\Leftarrow$" direction is trivial. On the other hand from $L = \mathrm{Spec}(\varphi)$ we can get that $L_1 = \mathrm{GenSpec}_{\varnothing}(\varphi')^*$ where φ' is Σ^1_1-formula which results from existentially quantifying every relational symbol in φ.

In the end we have $L \in \textsc{Spec} \Leftrightarrow \exists \mathcal{K} \in \textsc{GenSpec}_{\varnothing} : L_1 = \mathcal{K}^* \Leftrightarrow L_1 \in \mathbf{NP} \Leftrightarrow L \in \mathbf{NE}$, where the second "$\Leftrightarrow$" is just Fagin's theorem for $\tau = \varnothing$ because a tally language is just a set of codes of all $\varnothing$-structures of given sizes. □

3.4 $\mathbf{F}\Sigma^1_1$ operations on generalized spectra

In Chapter 2, we have shown that Spec is closed under union, intersection and many arithmetical operations. These operations can be, in some sense, understood as element-wise application of a function computable in polynomial-time. From the theorem 3.2.6 we have that all such functions are Σ^1_1-definable. In what sense are Σ^1_1-definable functions operations on spectra?

Again it is more natural to first consider generalized spectra. We now give sufficient conditions for Σ^1_1-function to be an operation on generalized spectra.

Theorem 3.4.1
Let $\mathcal{K}_1, \ldots, \mathcal{K}_k \in \textsc{GenSpec}$ in a vocabulary τ and $f \in \mathbf{F}\Sigma^1_1$ a k-ary function such that there exists $c \in \mathbb{N}^+$ and for each $i \in \{1, \ldots, k\}$ and $x_1, \ldots, x_k \in \{0,1\}^*$ we have $|x_i| \leq |f(x_1, \ldots, x_k)|^c$. Then

$$f(\mathcal{K}_1, \ldots, \mathcal{K}_k) = \{\mathcal{A}; \mathrm{code}_\tau(\mathcal{A}, <) \in f(\mathcal{K}^*_1, \ldots, \mathcal{K}^*_k)\} \quad (3.4.1)$$

is also a generalized spectrum in the vocabulary τ.

Proof. (sketch) We can define $f(\mathcal{K}_1, \ldots, \mathcal{K}_k)$ by existentially quantifying the input structures since the length of their codes is polynomially bounded by the length of the output code. From the proof of Theorem 3.3.1 we know that encoding of structures is Σ_1^1-definable. □

This theorem shows that, with assumptions on the input and ouput length, $\mathbf{F}\Sigma_1^1$ forms a large class of operations on generalized spectra. For $\tau = \varnothing$ this again translates to a statement about regular spectra. Namely each function f such that its unary version $f_1 \in \mathbf{F}\Sigma_1^1$ is an element-wise operation on spectra.

In Theorem 3.2.6 we proved that functions in $\mathbf{F}\Sigma_1^1$ contains every function computable in **FP**. Again, transitioning from unary, we get that functions computable in $\mathrm{DTime}(2^{\mathcal{O}(n)})$ have Σ_1^1-definable unary versions.

Corollary 3.4.2
Let $X_1, \ldots, X_k$ be spectra and f a k-ary function in $\mathrm{DTime}(2^{\mathcal{O}(n)})$ such that there exists $c \in \mathbb{N}^+$ such that $|x_i| \leq c \cdot |f(x_1, \ldots, x_k)|$ for every $i \in \{1, \ldots, k\}$. Then

$$f(X_1, \ldots, X_k) = \{f(x_1, \ldots, x_k); x_i \in X_i\} \in \textsc{Spec}. \tag{3.4.2}$$

Proof. Follows directly from Theorem 3.4.1 and the last paragraphs. □

Note, that there exist functions that, when applied element-wise, are operations on spectra yet their unary versions are not Σ_1^1-definable. Let $A \subseteq \mathbb{N}^+$ be non-recursive and define

$$f(x) = \begin{cases} 1 & x \in A \\ 2 & \text{otherwise.} \end{cases} \tag{3.4.3}$$

This function is clearly not recursive therefore f_1 cannot be Σ_1^1-definable, because every function in $\mathbf{F}\Sigma_1^1$ is recursive. This is because you can check for higher and higher values whether they satisfy the Σ_1^1-defining sentence in bigger and bigger structure. Also notice that f is an operation on Spec, because $\varnothing$, $\{1\}$, $\{2\}$ and $\{1, 2\}$ are all spectra.

Concluding remarks

In the first two chapters we introduced some notions we needed later on and gave some examples of spectra and operations on the class of all spectra that result in new spectra. In Chapter 3 we proved Fagin's theorem in a way that completely avoids arguments about formal machines by showing that every polynomial time function is Σ_1^1-definable; this notion was introduced in this context of our thesis. Lastly we have shown that a subclass of Σ_1^1-definable functions forms a class of operations on generalized spectra.

Apart from Asser's spectrum problem there are many other open problems regarding spectra. We again recommend [DJMM12] for a long list of open problems. Author of this thesis was particularly interested in the question whether categorical spectra CatSpec are equal to the complexity class **UE** [Fag93].

Categorical spectra are spectra of sentences that for each cardinality have at most one model up to isomorphism. Categorical generalized spectra are the classes of those models. Categorical generalized spectra are for example finite fields, linearly ordered sets and many others.

The complexity class **UE** (unambiguous exponential time) is the subclass of **NE** consisting of those languages which can be accepted in $\mathcal{O}(2^{c \cdot n})$ time and the accepting computation, if it exists, is unique. If we were to show that the codes of categorical generalized spectra CatGenSpec are exactly the languages in unambiguous polynomial time **UP** we could by the same argument as we proved the corollary 3.3.2 prove that CatSpec = **UE**.

This is not easy, because for a model M in a categorical generalized spectrum there could be actually many different codes. This is because the encoding depends on how is the structure ordered and there is not one canonical ordering. It is entirely possible that **UP** $\subsetneq$ CatGenSpec. Note that proving this is at least as hard as proving **P** $\neq$ **NP** since **P** $\subseteq$ **UP** and CatGenSpec $\subseteq$ GenSpec = **NP**.

Another interesting open problem is whether the set of all spectra over sentences with at most binary relations is equal to all spectra [DJMM12]. Notice that in the proofs of Theorems 3.2.4 and 3.2.5 we use relational symbols of arbitrary high arities and there does not seem to be a straightforward way to adapt the argument with just binary relational symbols.

Bibliography

[AB09] Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[Ass55] Günter Asser. Das repräsentenproblem in prädikatenkalkül der ersten stufe mit identität. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 1:252–263, 1955.

[Bos38] Raj Chandra Bose. On the application of the properties of galois fields to the problem of construction of hyper-graeco-latin squares. *Sankhyā: The Indian Journal of Statistics*, pages 323–338, 1938.

[CD01] Charles J. Colbourn and Jeffrey H. Dinitz. Mutually orthogonal latin squares: a brief survey of constructions. *Journal of Statistical Planning and Inference*, 95(1-2):9–48, 2001.

[Cob65] Alan Cobham. The intrinsic computational difficulty of functions. In Y. Bar-Hillel, editor, *Proceedings of the International Conference on Logic, Methodology, and Philosophy of Science*, pages 24–30. North Holland, 1965.

[DJMM12] Arnaud Durand, Neil D. Jones, Johann A. Makowsky, and Malika More. Fifty years of the spectrum problem: survey and new results. *The Bulletin of Symbolic Logic*, 18(4):505–553, 2012.

[Fag74] Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. In Richard Karp, editor, *Complexity of computation*, volume 7, page 2741. SIAM-ASM Proceedings, 1974.

[Fag93] Ronald Fagin. Finite-model theory-a personal perspective. *Theoretical computer science*, 116(1):3–31, 1993.

[GKL+07] Erich Grädel, Phokion G Kolaitis, Leonid Libkin, Maarten Marx, Joel Spencer, Moshe Y Vardi, Yde Venema, and Scott Weinstein. *Finite Model Theory and its applications*. Number 13 in Texts in Theoretical Computer Science (An EATCS Series). Springer-Verlag Berlin Heidelberg, 2007.

[JS74] Neil D Jones and Alan L. Selman. Turing machines and the spectra of first-order formulas. *The Journal of Symbolic Logic*, 39(1):139–150, 1974.

[Mar06] David Marker. *Model theory: an introduction*, volume 217 of *Graduate Texts in Mathematics*. Springer Science & Business Media, 2006.

[Rot10] Joseph J Rotman. *Advanced modern algebra*, volume 114 of *Graduate Texts in Mathematics*. American Mathematical Soc., 2010.

[Sch52] Heinrich Scholz. Ein ungelöstes problem in der symbolischen logik. *The Journal of Symbolic Logic*, 17:160, 1952.

[vdD10] Lou van den Dries. Mathematical logic lecture notes. https://faculty.math.illinois.edu/~vddries/, 2010.