On Uniformization in the Full Binary Tree

Alexander Rabinovich
The Blavatnik School of Computer Science, Tel Aviv University, Israel

Abstract

A function f uniformizes a relation R(X,Y) if R(X,f(X)) holds for every X in the domain of R. The uniformization problem for a logic L asks whether for every L-definable relation there is an L-definable function that uniformizes it. Gurevich and Shelah proved that no Monadic Second-Order (MSO) definable function uniformizes relation "Y is a one element subset of X" in the full binary tree. In other words, there is no MSO definable choice function in the full binary tree.

The cross-section of a relation R(X,Y) at D is the set of all E such that R(D,E) holds. Hence, a function that uniformizes R chooses one element from every non-empty cross-section. The relation "Y is a one element subset of X" has finite and countable cross-sections.

We prove that in the full binary tree the following theorems hold:

▶ **Theorem** (Finite cross-sections). *If every cross-section of an MSO definable relation is finite, then it has an MSO definable uniformizer.*

▶ **Theorem** (Uncountable cross-section). *There is an MSO definable relation R such that every MSO definable relation included in R and with the same domain as R has an uncountable cross-section.*

2012 ACM Subject Classification Theory of computation → Logic and verification

Keywords and phrases Monadic Second-Order Logic, Uniformization

Digital Object Identifier 10.4230/LIPIcs.MFCS.2022.77

1 Introduction

Let $R \subseteq D_1 \times D_2$ be a binary relation. For $d_1 \in D_1$ the cross-section of R at d_1 is the set $R_{d_1} := \{d \in D_2 \mid (d_1, d) \in R\}$. The domain of R is the set $\{d_1 \in D_1 \mid \exists d_2 (d_1, d_2) \in R\}$.

A uniformizer of R is a subset R^* of R such that for all x: $\exists y R(x,y) \Leftrightarrow \exists! y R^*(x,y)$ (where $\exists!$ stands for "there exists unique"). Hence, a uniformizer for R is a partial function that chooses an element from each non-empty cross-section of R and has the same domain as R (see Fig. 1).

In other words, given an input x for which the original relation R has a non-empty cross-section, a uniformizer returns a single value from the cross-section at x. This is a special case of a choice function.

The axiom of choice implies that a uniformizer always exists; however, it is often important that a uniformizer has some "nice" properties. This is where the uniformization theorems come into action. They guarantee that, if R satisfies certain properties, then it has a uniformizer that has other desirable properties.

A set of relations $\mathfrak{R}$ has the uniformization property if every $R \in \mathfrak{R}$ has a uniformizer in $\mathfrak{R}$.

We consider the set of relations over the full binary tree definable in the Monadic Second-Order Logic (MSO). The seminal Rabin's theorem states that the Monadic Second-Order Logic is decidable over the full binary tree [8].

The monadic second-order logic is an extension of first-order logic by set variables (which range over the subsets of the domain of a structure), and the quantifiers over the set variables (see Section 2.1).

47th International Symposium on Mathematical Foundations of Computer Science (MFCS 2022).
Editors: Stefan Szeider, Robert Ganian, and Alexandra Silva; Article No. 77; pp. 77:1–77:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

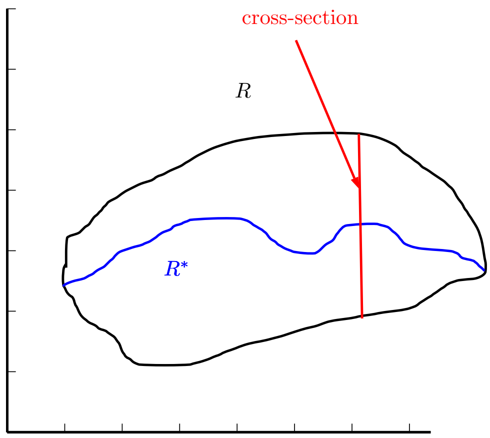

Figure 1 R^* uniformizes R.

An MSO formula $\alpha(X, Y)$ with free set (second-order) variables X and Y defines a binary relation on the subsets of the full binary tree. Rabin [8] proved the basis theorem that states: every non-empty relation definable by an MSO formula $\alpha(Y)$ contains an MSO definable set. The basis theorem is a simple (degenerate) instance of the uniformization property. Rabin asked whether the class of relations definable in the monadic second-order logic over the full binary tree has the uniformization property. Gurevich and Shelah [6] gave a negative solution to Rabin's question with the formula $\alpha(X, Y)$ saying "if X is non-empty, then Y is a one element subset of X" as a counter-example. A function that uniformizes "Y is a one element subset of X" is a choice function; it chooses one element from every non-empty set X. The Gurevich-Shelah Theorem states that there is no MSO definable choice function in the full binary tree.

Since MSO definable relations over the tree do not have the uniformization property, a natural task is to decide whether an MSO definable relation has a (MSO-definable) uniformizer; another natural task is to provide a characterization of those relations which have a uniformizer. The decidability whether a relation has a uniformizer is open. However, we provide a sufficient condition for a relation to have a uniformizer.

▶ **Theorem** (Finite cross-sections). *If the cross-sections of an MSO definable relation over the full binary tree are finite, then it has an MSO definable uniformizer.*

Note that the Gurevich-Shelah Theorem and the above Theorem imply that there are MSO definable relations that do not include MSO definable relations with the same domain and only finite cross-sections.

A natural question is whether an MSO definable relation always includes an MSO definable relation with the same domain and at most countable cross-sections. We provide a negative answer to this question.

▶ **Theorem** (Uncountable cross-section). *There is an MSO definable relation R such that any MSO definable relation which is included in R and has the same domain as R has an uncountable cross-section.*

Hence, it is even impossible to choose in a definable way a countable subset from every non-empty cross-section.

The paper is organized as follows. Section 2 recalls standard definitions about monadic second-order logics, trees, and presents elements of composition method which are used in our proofs. Section 4 provides a proof of Finite cross-section theorem. Section 5 discusses the consequences of undefinability of choice function; it also introduces some notions which are helpful for the proof of Uncountable cross-section theorem given in Section 6. Section 7 contains conclusions and further results.

2 Preliminaries

We use standard notations and terminology. We use n, k, m, l for natural numbers. We use capital letters A, B, C for sets, and lower case letters a, b, c for elements of sets. The powerset of a set D is denoted by $\mathbb{P}(D)$. We use the expressions "chain" and "linear order" interchangeably; we use ω for the order type of the natural numbers.

2.1 Monadic Second-Order Logic

We use standard notations and terminology about Monadic Second-Order logic (MSO) [8, 12, 10].

Let σ be a relational signature. A structure (for σ) is a tuple $\mathcal{M} = (D, \{R^{\mathcal{M}} \mid R \in \sigma\})$ where D is a domain, and each symbol $R \in \sigma$ is interpreted as a relation $R^{\mathcal{M}}$ on D.

MSO formulas use first-order variables, which are interpreted as elements of the structure, and monadic second-order variables, which are interpreted as sets of elements. Atomic MSO formulas are of the following form:

- $R(x_1, \dots, x_n)$ for an n-ary relational symbol R and first-order variables $x_1, \dots, x_n$
- $x = y$ for two first-order variables x and y
- $x \in X$ for a first-order variable x and a second-order variable X

MSO formulas are constructed from the atomic formulas, using boolean connectives, the first-order quantifiers, and the second-order quantifiers.

We write $\psi(X_1, \dots, X_n, x_1, \dots, x_m)$ to indicate that the free variables of the formula ψ are $X_1, \dots, X_n$ (second-order variables) and $x_1, \dots, x_m$ (first-order variables). We write $\mathcal{M}, A_1, \dots, A_n, a_1, \dots, a_m \models \psi$ or $\mathcal{M} \models \psi(A_1, \dots, A_n, a_1, \dots a_m)$ if ψ holds in $\mathcal{M}$ when subsets A_i are assigned to X_i for $i = 1, \dots, n$ and elements a_i are assigned to variables $x_1, \dots, x_m$ for $i = 1, \dots, m$. Whenever $\mathcal{M}$ is clear from the context we will further abbreviate this to $\psi(A_1, \dots, A_n, a_1, \dots a_m)$. Sometimes, we abuse notations and use X for a variable and for the set assigned to it.

▶ **Definition 2.1** (Definability). *Let $\psi(X_1, \dots, X_n)$ be an MSO formula and $\mathcal{M}$ a structure. The relation defined by ψ in $\mathcal{M}$ is the set*

$$\psi\mathcal{M} := \{(D_1, \dots, D_n) \in \mathbb{P}(D)^n \mid \mathcal{M} \models \psi(D_1 \dots, D_n)\}.$$

A relation is MSO-definable in $\mathcal{M}$ if it is equal to $\psi\mathcal{M}$ for an MSO formula ψ. A set $U \subseteq D$ is MSO-definable in $\mathcal{M}$ if there is a formula $\psi(X)$ such that $\psi\mathcal{M} = \{U\}$. A function is MSO-definable in $\mathcal{M}$ if its graph is.

Let $\psi(X_1, \dots, X_n, Y_1, \dots Y_l)$ be an MSO formula and $\mathcal{M}$ a structure. Let $\overline{C}$ be an l-tuple of subsets of the domain D of $\mathcal{M}$. The relation defined by ψ in $\mathcal{M}$ with parameters *$\overline{C}$ is the set*

$$\psi(\mathcal{M}, \overline{C}) := \{(D_1, \dots, D_n) \in \mathbb{P}(D)^n \mid \mathcal{M} \models \psi(D_1 \dots, D_n, \overline{C}\}.$$

The definability of a subset and of a function in $\mathcal{M}$ with parameters is defined in a similar way.

2.2 Trees

We view the set $\{l,r\}^*$ of finite words over the alphabet $\{l,r\}$ as the domain of the full binary tree, where the empty word ϵ is the root of the tree, and for each node $v \in \{l,r\}^*$, we call $v \cdot l$ the left child of v, and $v \cdot r$ the right child of v.

We define a tree order "$\leq$" as a partial order such that $\forall u, v \in \{l,r\}^* : u \leq v$ iff u is a prefix of v.

Nodes u and v are incomparable - denoted by $u \perp v$ - if neither $u \leq v$ nor $v \leq u$; a set U of nodes is an **antichain**, if its elements are incomparable with each other.

We say that an infinite sequence $\pi = v_0, v_1, \ldots$ is a **tree branch** if $v_0 = \epsilon$ and $\forall i \in \mathbb{N}$: $v_{i+1} = v_i \cdot l$ or $v_{i+1} = v_i \cdot r$.

A path is a finite or infinite sequence $v_0 \ldots v_i \ldots$ such that if v_i is not the last node, then $v_{i+1} = v_i \cdot l$ or $v_{i+1} = v_i \cdot r$.

We consider the full binary tree as a structure for a signature $\{\leq, left, right\}$ where unary relation symbols $left$ and $right$ are interpreted as $\{wl | w \in \{l,r\}^*\}$ and $\{wr | w \in \{l,r\}^*\}$, respectively; $\leq$ is interpreted as the prefix relation.

A k-tree is an expansion of the full binary tree by k unary predicates. Whenever k is clear from the context or unimportant we will use "labelled tree" for "k-tree."

Given a k-tree $\mathfrak{T} := (T, P_1, \ldots, P_k)$ and a node v in $\mathfrak{T}$, the k-tree $\mathfrak{T}_{\geq v} := (T_{\geq v}, P'_1, \ldots, P'_k)$ (called the subtree of $\mathfrak{T}$, rooted at v) is defined by $T_{\geq v}$ is the full binary tree, and $u \in P'_i$ iff $vu \in P_i$ for $i = 1, \ldots, k$.

Let F be an antichain in a k-tree $\mathfrak{T}$ and let $\mathfrak{T}_1$ be a k-tree. The **grafting** of $\mathfrak{T}_1$ at F in $\mathfrak{T}$ is denoted by $\mathfrak{T}\{\mathfrak{T}_1/F\}$ and is the k-tree obtained from $\mathfrak{T}$ when the subtrees rooted at F are replaced by $\mathfrak{T}_1$.

Formally, $\mathfrak{T}' := \mathfrak{T}\{\mathfrak{T}_1/F\}$ is defined as follows: let $u \in \{l,r\}^*$ be a node. (i) if $\neg\exists f \in F(f \leq u)$ then $P_i^{\mathfrak{T}'}(u)$ iff $P_i^{\mathfrak{T}}(u)$; else (ii) $u = fv$ for a unique $f \in F$, and $P_i^{\mathfrak{T}'}(u)$ iff $P_i^{\mathfrak{T}_1}(v)$.

2.3 Composition Method

The proofs presented in this paper make use of the technique known as the "composition method." To fix notations and to aid the reader not familiar with this technique, we briefly review those definitions and results that we need. A more detailed presentation can be found in [11] or [5, 7].[1]

Let σ be a finite relational signature. Write σ_k for the signature σ with k (fresh) unary predicate $P_1, \cdots, P_k$ symbols. Thus, a σ_k-structure $\mathfrak{A}$ has the form $(A, P_1^{\mathfrak{A}}, \cdots, P_k^{\mathfrak{A}})$, where A is a σ-structure. The quantifier rank of a formula φ, denoted $\mathrm{qr}(\varphi)$, is the maximum depth of the nesting of quantifiers in φ. For example, if ψ and φ are quantifier-free, then the quantifier rank of $(\exists x \forall Y \forall Z \psi) \wedge (\forall Y \exists u \varphi)$ is 3. For $r, k \in \mathbb{N}$ we denote by $\mathfrak{Form}_k^r$ the set of formulas of quantifier rank $\leq r$ and with free variables among $X_1, \ldots, X_k$ in signature σ.

For σ_k-structures $\mathfrak{A}, \mathfrak{B}$ write $\mathfrak{A} \equiv_k^r \mathfrak{B}$ if for every $\varphi \in \mathfrak{Form}_k^r$,

$$\mathfrak{A} \models \varphi(P_1^{\mathfrak{A}}, \cdots, P_k^{\mathfrak{A}}) \text{ if and only if } \mathfrak{B} \models \varphi(P_1^{\mathfrak{B}}, \cdots, P_k^{\mathfrak{B}}).$$

Clearly, $\equiv_k^r$ is an equivalence relation and the set $\mathfrak{Form}_k^r$ is infinite. Since the signature σ_k is finite and relational, the set $\mathfrak{Form}_k^r$ contains only finitely many semantically distinct formulas, so there are only finitely many $\equiv_k^r$-classes of σ_k-structures. The following lemma isolates maximally consistent formulas.

[1] In [9], [6], and several other papers, the technique is further developed and a much deeper application of it is made than will be made here.

▶ **Lemma 2.2** (Hintikka lemma). *Let σ be a finite relational signature. For $r, k \in \mathbb{N}$, there is a* finite *set $H^r_k \subseteq \mathfrak{Form}^r_k$ such that:*
1. *For every σ_k-structure $\mathfrak{A}$ there is a* unique *$\tau \in H^r_k$ with $\mathfrak{A} \models \tau(P^{\mathfrak{A}}_1, \cdots, P^{\mathfrak{A}}_k)$.*
2. *If $\tau \in H^r_k$ and $\varphi \in \mathfrak{Form}^r_k$, then $\tau \models \varphi$ or $\tau \models \neg\varphi$.*[2]

Elements of H^r_k are called (r, k)-*Hintikka formulas.*

▶ **Definition 2.3** (type of a structure). *For a σ_k-structure $\mathfrak{A}$, write* $\text{type}^r_k(\mathfrak{A})$ *for the unique $\tau(X_1, \cdots, X_k) \in H^r_k$ such that $\mathfrak{A} \models \tau(P^{\mathfrak{A}}_1, \cdots, P^{\mathfrak{A}}_k)$, and call it the* (r, k)-type *of $\mathfrak{A}$.*

Thus, $\text{type}^r_k(\mathfrak{A})$ determines for which formulas $\varphi \in \mathfrak{Form}^r_k$ it holds that $\mathfrak{A} \models \varphi(P^{\mathfrak{A}}_1, \cdots, P^{\mathfrak{A}}_k)$. Since k is often clear, we may drop it, and write $\text{type}^r(\mathfrak{A})$ and call it the r-type of $\mathfrak{A}$.

We now state weak versions of the composition theorem for MSO over chains and trees, see [9] or [6, 5] for details. The first deals with a representation of ω-chains as a concatenation of finite chains; the second considers a branch in a tree.

▶ **Lemma 2.4** (Weak Composition Theorem for ω-chains). *Let $(L, <)$ be a linear order isomorphic to ω. Let $v_1 < v_2 < \cdots < v_n < \cdots$ be a sequence of elements in L, where v_1 is the minimal element of L. Let $\mathcal{L} := (L, <, P_1, \cdots, P_k)$ be an expansion of $(L, <)$ by unary predicates, and let $\mathcal{L}_i$ be the substructure of $\mathcal{L}$ over $\{v \mid v_i \leq v < v_{i+1}\}$. Then, $\text{type}^m(\mathcal{L})$ is determined by the sequence $\text{type}^m(\mathcal{L}_i)$ $(i = 1, \dots)$.*

▶ **Lemma 2.5** (Weak Composition Theorem for a Tree Branch). *Let T be the full binary tree and let P_1, P_2, Q_1, Q_2 be subsets of the nodes of T. Let $T^1 := (T, P_1, Q_1)$ and $T^2 := (T, P_2, Q_2)$ be the expansion of T by the unary predicates. Let $\pi := v_1 v_2 \dots$ be a branch in T. Let u_i be a child of v_{i-1} different from v_i. Let $\tau^i_1 := \text{type}^n(T^1_{\geq u_i})$ and $\tau^i_2 := \text{type}^n(T^2_{\geq u_i})$. Assume that $\forall i. \tau^i_1 = \tau^i_2$ and $\pi \cap P_1 = \pi \cap P_2$ and $\pi \cap Q_1 = \pi \cap Q_2$. Then $\text{type}^n(T^1) = \text{type}^n(T^2)$.*

A lemma similar to Lemma 2.5 holds when pairs $\langle P_i, Q_i \rangle$ are replaced by tuples $\langle P_i, Q_i, R_i, \cdots \rangle$.

3 Uniformization

▶ **Definition 3.1** (Uniformization). *Let $\varphi(\overline{X}, \overline{Y})$, $\psi(\overline{X}, \overline{Y})$ be formulas and $\mathcal{C}$ a class of structures. We say that ψ* uniformizes *(or is a* uniformizer *for) φ* over *$\mathcal{C}$ iff for all $\mathcal{M} \in \mathcal{C}$:*
1. *$\mathcal{M} \models \forall \overline{X} \exists^{\leq 1} \overline{Y} \psi(\overline{X}, \overline{Y})$,*
2. *$\mathcal{M} \models \forall \overline{X} \forall \overline{Y} (\psi(\overline{X}, \overline{Y}) \to \varphi(\overline{X}, \overline{Y}))$, and*
3. *$\mathcal{M} \models \forall \overline{X} \big(\exists \overline{Y} \varphi(\overline{X}, \overline{Y}) \to \exists \overline{Y} \psi(\overline{X}, \overline{Y})\big)$.*

Here, $\overline{X}, \overline{Y}$ are tuples of distinct variables and "$\exists^{\leq 1} \overline{Y} \dots$" stands for "there exists at most one ...". Hence, the first item says that ψ is a graph of a partial function.

The class $\mathcal{C}$ is said to have the uniformization property *iff every formula φ has a uniformizer ψ over $\mathcal{C}$.*

If $\mathcal{C} = \{\mathcal{M}\}$ consists of a single structure, we speak of uniformization in *$\mathcal{M}$ rather than* over *$\mathcal{C}$.*

First, note that, strictly speaking, the question whether φ uniformizes ψ over $\mathcal{C}$ depends not only on formulas φ and ψ, but on the formulas together with a partition of their free-variables into domain variables $\overline{X}$ and image variables $\overline{Y}$. In the few cases where we use

[2] Furthermore, H^r_k is computable from r, k, and there is an algorithm that given τ and φ decides between $\tau \models \varphi$ and $\tau \models \neg\varphi$. We do not use these facts.

letters other than X and Y, we shall state explicitly which variables are to be taken as domain variables and which as image variables and say that φ uniformizes ψ for $(\overline{U};\overline{V})$ over $\mathcal{C}$ if $\overline{U}$ is the set of domain variables and $\overline{V}$ is the set of image variables.

If the cross-sections of a relation R are finite, we say that R is a *finitary* relation.

▶ **Lemma 3.2** (Reducing image variables)**.** *Assume that every finitary relation in $\mathcal{M}$ definable by an* MSO *formula with one image variable has an* MSO *definable uniformizer. Then, every* MSO *definable finitary relation in $\mathcal{M}$ has an* MSO *definable uniformizer.*

4 Finite Cross-sections

▶ **Theorem 4.1** (Finite cross-sections)**.** *If the cross-sections of an MSO definable relation over the full binary tree are finite, then it has an MSO definable uniformizer.*

▶ Remark 4.2 (On computability). It is decidable whether the cross sections of the relation definable by an MSO formula are finite. Our proof of Theorem 4.1 is constructive, and an algorithm which provides an MSO formula that defines a uniformizer can be easily extracted from the proof.

By Lemma 3.2, it is sufficient to prove Theorem 4.1 for the relations definable by formulas with one image variable. To simplify notations we consider formulas with one domain and one image variables. However, everywhere in the proof a domain variable can be replaced by a tuple of domain variables.

▶ **Notations 4.3.** *For a formula $\alpha(X, Y)$ and a subset P of the full binary tree we denote*

1. $\Im\alpha(P) := \{Q \mid \alpha(P, Q)\}$ *- the α-image of P (this is the cross-section of the relation defined by α at P).*
2. $\max_{\subseteq} \Im\alpha(P) := \{Q \in \Im\alpha(P) \mid \neg\exists Y' \in \Im\alpha(P)(Q \subsetneq Y')$ *- $\subseteq$-maximal elements of* $\Im\alpha(P)\}$*. This set is non-empty when $\Im\alpha(P)$ is finite and non-empty.*

It is clear that $\Im\alpha(P)$ and $\max_{\subseteq} \Im\alpha(P)$ are MSO-definable with parameter P.

▶ **Open Question.** *Does there exist an MSO definable linear order on the subsets of the full binary tree?*

If the answer to the question is positive, we can define a uniformizer for α which for every X returns a minimal element in $\Im\alpha(X)$. We have not succeeded to answer the question. However, we still can construct an MSO-definable uniformizer for α. If $\max_{\subseteq} \Im\alpha(P)$ has only one element, the uniformizer chooses it. In Subsection 4.1 we will show how to choose in a definable way between two elements. Relying on this construction, we show in Subsection 4.2 how to choose in a definable way from a finite $\max_{\subseteq} \Im\alpha(P)$, and therefore, from any finite cross-section.

4.1 Choose one Set from two Sets

Here, we are going to show how to choose in a definable way between two elements $Q_1, Q_2 \in \max_{\subseteq} \Im\alpha(P)$.

Let $U_1 := Q_1 \setminus Q_2$ and $U_2 := Q_2 \setminus Q_1$. U_1 and U_2 are non-empty and disjoint. We will use $v\uparrow$ for $\{u \mid v \leq u\}$.

Generate a path π as described in **Procedure** Generate π on page 7. We start from the root and at every iteration extend the path by one node. The procedure maintains the invariant: if u is not the last node on the path, then $u\uparrow \cap U_1 \neq \emptyset \neq u\uparrow \cap U_2$ holds. The generated path might be finite (in this case the procedure returns from line 4, 7 or 10) or infinite.

Procedure Generate π.

```
u ← root; π ← {u}
while true do
    if u ∈ U1 ∪ U2 then
        return                                                        // (a)
    else if ul↑ has a non-empty intersection exactly with one of U1, U2 then
        π ← π ∪ {ul}
        return                                                        // (b)
    else if ur↑ has a non-empty intersection exactly with one of U1, U2 then
        π ← π ∪ {ur}
        return                                                        // (b)
    else if ul↑ ∩ U1 ≠ ∅ ≠ ul↑ ∩ U2 then
        u ← ul
        π ← π ∪ {u}
    else
        /* in this case ur↑ ∩ U1 ≠ ∅ ≠ ur↑ ∩ U2                        */
        u ← ur
        π ← π ∪ {u}
```

It is clear that π is MSO-definable with parameters U_1 and U_2, i.e., there is an MSO formula $\mu(X_1, X_2, Z)$ such that $\mu(U_1, U_2, \pi)$ holds iff π is generated by this procedure. Moreover, the formula is symmetrical in the parameters, i.e., $\mu(U_1, U_2, \pi) \leftrightarrow \mu(U_2, U_1, \pi)$.

Abbreviations. Below we will abbreviate $\text{type}^n(T, P, Q)$ as $\text{type}^n(P, Q)$ and $\text{type}^n(T, P, Q)_{\geq v}$ as $\text{type}^n(P, Q)_{\geq v}$.

Let us analyse what happens if $\pi := u_1 u_2 \ldots$ is infinite. In this case for every i: $u_i \notin (U_1 \cup U_2)$ and $u_i{\uparrow} \cap U_1 \neq \emptyset \neq u_i{\uparrow} \cap U_2$.

Let v_i be a child of u_{i-1} different from u_i. Let n be the quantifier rank of α. Let $\tau_1^i := \mathit{type}^n(P, Q_1)_{\geq v_i}$ and $\tau_2^i := \mathit{type}^n(P, Q_2)_{\geq v_i}$.

▷ Claim 4.4. If π is infinite, then $\exists i(\tau_1^i \neq \tau_2^i)$.

Proof. Toward a contradiction we assume $\forall i(\tau_1^i = \tau_2^i)$, and derive that $\exists^{2^{\aleph_0}} Y \alpha(P, Y)$.

Indeed, let $D := \{i \in \mathbb{N} \mid U_1 \cap (v_i{\uparrow}) \neq \emptyset\}$. First, we observe that D is infinite. Indeed, assume that D is finite and i_0 is its maximal element. Note that $(u_i{\uparrow} \cap U_1) = \{u \in \pi \cap U_1 \mid u \geq u_i\} \cup \bigcup_{j>i}(v_j{\uparrow} \cap U_1)$. Since $\pi \cap U_1 = \emptyset$ and $v_j{\uparrow} \cap U_1 = \emptyset$ for $i > i_0$, we conclude that $u_i{\uparrow} \cap U_1 = \emptyset$ for $i > i_0$. A contradiction.

Now, for every $I \subseteq D$, define $Q_I := \bigcup_{i \in I}(v_i{\uparrow} \cap Q_1) \cup \bigcup_{i \notin I}(v_i{\uparrow} \cap Q_2)$. For every $i \in \mathbb{N}$:

$$\tau_1^i = \mathit{type}^n(P, Q_1)_{\geq v_i} = \tau_2^i = \mathit{type}^n(P, Q_2)_{\geq v_i} = \mathit{type}^n(P, Q_I)_{\geq v_i}.$$

Therefore, by Lemma 2.5

$$\mathit{type}^n(P, Q_1) = \mathit{type}^n(P, Q_I \cup (Q_1 \cap \pi)) \text{ for every } I.$$

Since $\alpha(P, Q_1)$ and the quantifier rank of α is n, we obtain that

$$\alpha(P, Q_I \cup (Q_1 \cap \pi)) \text{ for every } I.$$

For $I_1 \neq I_2 \subseteq D$ we have $Q_{I_1} \neq Q_{I_2}$; moreover, both Q_{I_1} and Q_{I_2} have the empty intersection with π. Therefore, $(Q_{I_1} \cup (Q_1 \cap \pi)) \neq (Q_{I_2} \cup (Q_1 \cap \pi))$, for $I_1 \neq I_2 \subseteq D$. Hence, $|\Im\alpha(P)| \geq |\{I \mid I \subseteq D\}| = 2^{\aleph_0}$. This contradicts our assumption that the cross-sections are finite.

Hence, if π is infinite, then there is $v_i \in \pi$ such that $\tau_1^i \neq \tau_2^i$. ◁

For every P and $Q_1 \neq Q_2 \in \max_{\subseteq} \Im\alpha(P)$ let π be the corresponding path. Define

$$G(P, Q_1, Q_2) = \begin{cases} \text{the last element on } \pi & \text{if } \pi \text{ is finite} \\ \text{the minimal } u_i \in \pi \text{ with } \tau_1^i \neq \tau_2^i & \text{otherwise} \end{cases}$$

From Claim 4.4 and definability of π, we derive that G is total and an MSO definable function.

The next Lemma summarizes properties of a formula which defines G.

▶ **Lemma 4.5.** *There is $\psi(X, Y_1, Y_2, z)$ such that*

1. $\psi(X, Y_1, Y_2, z) \leftrightarrow \psi(X, Y_2, Y_1, z)$
2. $(\psi(X, Y_1, Y_2, z) \wedge Y_1, Y_2 \in \max_{\subseteq} \Im\alpha(X) \wedge Y_1 \neq Y_2) \to \exists! z \psi(X, Y_1, Y_2, z)$
3. *if $\psi(X, Y_1, Y_2, z) \wedge Y_1, Y_2 \in \max_{\subseteq} \Im\alpha(X) \wedge Y_1 \neq Y_2$, then one of the following conditions holds:*
   a. *$z \in Y_1 \Delta Y_2$ (z in the symmetric difference of Y_1 and Y_2),*
   b. *let $U_1 := Y_1 \setminus Y_2$ and $U_2 := Y_2 \setminus Y_1$, then $U_1 \cap z\!\uparrow \neq \emptyset \wedge U_2 \cap z\!\uparrow = \emptyset$, or $U_1 \cap z\!\uparrow = \emptyset \wedge U_2 \cap z\!\uparrow \neq \emptyset$,*
   c. *$type^n(X, Y_1)_{\geq z} \neq type^n(X, Y_2)_{\geq z}$.*

Now, we are ready to explain how to choose in a definable way between two elements $Y_1, Y_2 \in \max_{\subseteq} \Im\alpha(X)$. We are going to choose according to the cases (a)-(c) of Lemma 4.5(3). If 3(a) holds, then if $z \in Y_1$ choose Y_1 else choose Y_2. If 3(a) fails, but 3(b) holds, then if $U_1 \cap z\!\uparrow \neq \emptyset$ choose Y_1 else choose Y_2. Let $\leq_n$ be any linear order on a finite set of n-types. If 3(a) and 3(b) fail, then 3(c) holds. In this case, if $type^n(X, Y_1)_{\geq z} <_n type^n(X, Y_2)_{\geq z}$ choose Y_1 else Y_2.

4.2 Choose a Set from Finitely Many Sets

Below we explain how to choose from $\max_{\subseteq} \Im\alpha(P)$ when its cardinality is finite and > 2.

Let $F(X, Y) := \{z \mid (\exists Y' \in \max_{\subseteq} \Im\alpha(X))\psi(X, Y, Y', z)\}$.

F is an MSO-definable function. $F(X, Y)$ maps $\max_{\subseteq} \Im\alpha(P)$ to finite non-empty sets.

Recall that the lexicographical order $<_{lex}$ is an MSO-definable linear order on the nodes of the full binary tree. Define a linear order $\prec_{lex}$ on the finite sets as: $Z' \prec_{lex} Z$ if there is $z \in Z \setminus Z'$ such that $\forall y <_{lex} z (y \in Z \leftrightarrow y \in Z')$. It is clear that $\prec_{lex}$ is MSO-definable.

Assume $Y_1 \neq Y_2 \in \max_{\subseteq} \Im\alpha(X)$ and $F(X, Y_1) = F(X, Y_2) = Z$. Let $z := G(X, Y_1, Y_2)$. Then, $z \in Z$ and one of the conditions 3(a)-3(c) of Lemma 4.5 holds.

Now let us explain how to choose Y from $\max_{\subseteq} \Im\alpha(X)$. It will be easy to see that all the sets and relations described below are MSO-definable. (We will give a verbal description of the relation, leaving it to the reader to check that this verbal description is expressible by an MSO formula.)

1. Let $Z_{\min} := \min_{\prec_{lex}}\{Z = F(X, Y) \mid Y \in \max_{\subseteq} \Im\alpha(X)\}$. We will choose Y from $\Gamma := \{Y \in \max_{\subseteq} \Im\alpha(X) \mid F(X, Y) = Z_{\min}\}$.
2. Define (linear) pre-orders $<_A$, $<_B$ and $<_C$ on Γ (these correspond to items 3(a), 3(b) and 3(c) of Lemma 4.5).

   $Y_1 <_A Y_2$ if $(Y_1 \cap Z_{\min}) \prec_{lex} (Y_2 \cap Z_{\min})$

Let $\Gamma_A :=$ the set of $<_A$ minimal elements of Γ. It is easy to see that if $Y_1, Y_2 \in \Gamma_A$, $Y_1 \neq Y_2$ and $\psi(X, Y_1, Y_2, z)$, then condition 3(a) of Lemma 4.5 fails. Indeed, if 3(a) holds, then $z \in Z_{\min}$ and $z \in Y_1 \Delta Y_2$. However, the minimality of Γ_A implies that $(Y_1 \cap Z_{\min}) = (Y_2 \cap Z_{\min})$ for every $Y_1, Y_2 \in \Gamma_A$. Contradiction.

Now define $<_B$ on Γ_A. Let $U_1 := Y_1 \setminus Y_2$ and $U_2 := Y_2 \setminus Y_1$,

$$Y_1 <_B Y_2 \text{ if } \exists z \in Z_{\min} \text{ such that } U_2 \cap z{\uparrow} \neq \emptyset \wedge U_1 \cap z{\uparrow} = \emptyset \wedge \\ \forall z' \in Z_{\min}\ z' <_{lex} z \rightarrow (U_1 \cap z'{\uparrow} = \emptyset) \leftrightarrow (U_2 \cap z'{\uparrow} = \emptyset)$$

Let $\Gamma_B :=$ the set of $<_B$ minimal elements of Γ_A. It is easy to see that if $Y_1, Y_2 \in \Gamma_B$, and $Y_1 \neq Y_2$ and $\psi(X, Y_1, Y_2, z)$, then condition 3(b) of Lemma 4.5 fails.

Recall that $\leq_n$ is a linear order on a finite set of n-types. Define $<_C$ on Γ_B.

$$Y_1 <_C Y_2 \text{ if } \exists z \in Z_{\min} \text{ such that} \\ \mathit{type}^n(X, Y_1)_{\geq z} <_n \mathit{type}^n(X, Y_2)_{\geq z} \wedge \\ \forall z' \in Z_{\min}\ z' <_{lex} z \rightarrow (\mathit{type}^n(X, Y_1)_{\geq z'} = \mathit{type}^n(X, Y_2)_{\geq z'})$$

Observe

▷ Claim 4.6. $<_C$ is a linear order on Γ_B.

Proof. It is clear that $<_C$ is irreflexive. We will prove that (1) $<_C$ is transitive and (2) If $Y_1 \neq Y_2$ then $Y_1 <_C Y_2$ or $Y_2 <_C Y_1$. These imply that $<_C$ is a linear order.

(1) $<_C$ is transitive. Indeed, let $Y_1, Y_2, Y_3 \in \Gamma_B$ and $Y_1 <_C Y_2 <_C Y_3$. Assume that $z_{1,2}$ is a witness that $Y_1 <_C Y_2$, i.e., $\mathit{type}^n(X, Y_1)_{\geq z_{1,2}} <_n \mathit{type}^n(X, Y_2)_{\geq z_{1,2}}$ and $\forall z' \in Z_{\min}\ z' <_{lex} z_{1,2} \rightarrow (\mathit{type}^n(X, Y_1)_{\geq z'} = \mathit{type}^n(X, Y_2)_{\geq z'}$. Assume that $z_{2,3}$ is a witness that $Y_2 <_C Y_3$.

If $z_{1,2} <_{lex} z_{2,3}$, then $z_{1,2}$ is a witness that $Y_1 <_C Y_3$; otherwise $z_{2,3}$ is a witness that $Y_1 <_C Y_3$. Hence, $<_C$ is transitive.

(2) Now, we prove that If $Y_1 \neq Y_2$ then $Y_1 <_C Y_2$ or $Y_2 <_C Y_1$.

Since, $Y_1, Y_2 \in \Gamma_B$, we know that

if $\psi(X, Y_1, Y_2, z)$, then 3(a) and 3(b) of Lemma 4.5 fail. Hence, 3(c) holds, i.e., $\mathit{type}^n(X, Y_1)_{\geq z} \neq \mathit{type}^n(X, Y_2)_{\geq z}$ Therefore,

$$\exists z \in Z_{\min} \text{ such that } \mathit{type}^n(X, Y_1)_{\geq z} \neq \mathit{type}^n(X, Y_2)_{\geq z} \wedge \\ \forall z' \in Z_{\min}\ z' <_{lex} z \rightarrow (\mathit{type}^n(X, Y_1)_{\geq z'} = \mathit{type}^n(X, Y_2)_{\geq z'}$$

Hence, $Y_1 <_C Y_2$ or $Y_2 <_C Y_1$. ◁

Now, we choose (a unique) $<_C$-minimal element of Γ_B.

Below is the summary of our choice of $Y \in \max_{\subseteq} \Im\alpha(X)$:

1. $Z_{\min} := \min_{\prec_{lex}}\{Z = F(X, Y) \mid Y \in \max_{\subseteq} \Im\alpha(X)\}$ and $\Gamma := \{Y \in \max_{\subseteq} \Im\alpha(X) \mid F(X, Y) = Z_{\min}\}$.
2. $\Gamma_A :=$ the set of $<_A$ minimal elements of Γ.
3. $\Gamma_B :=$ the set of $<_B$ minimal elements of Γ_A.
4. We choose a unique $<_C$ minimal elements of Γ_B.

It is clear that the above choice can be formalized in MSO.

5 Choice Function and Fooling Sets

A *choice function* is a mapping which assigns to each non-empty set of nodes one element from the set.

Gurevich and Shelah [6] proved:

▶ **Theorem 5.1** (Gurevich and Shelah [6]). *There is no MSO-definable choice function in the full-binary tree.*

A simplified combinatorial proof of Theorem 5.1 was given in [2, 3].

In the rest of this section we introduce some notions and prove lemmas which will be used in the next section to prove uncountable cross-section theorem.

Choice Function on antichains. A choice function on antichains is a mapping which assigns to each non-empty antichain in the full binary tree one element from the antichain.

▶ **Corollary 5.2.** *There is no MSO-definable choice function on antichains.*

Proof. If $\beta(X, y)$ defines a choice function on antichains then $\alpha(X, y) := \exists Z$ "Z is the set of $\leq$ minimal elements of X"$\wedge\beta(Z, y)$ defines a choice function - a contradiction. ◀

▶ **Definition 5.3** (Fooling set). *Assume $\alpha(X, y) \rightarrow y \in X$. A set P is fooling for $\alpha(X, y)$ (wrt choice) if $P \neq \emptyset$ and $\neg\exists! y\alpha(P, y)$. P is a fooling set for a set Φ of formulas if it is a fooling set for each formula in Φ. If, in addition, P is an antichain, we say that P is a fooling antichain.*

It is helpful to use the following convention: If $\alpha(X, y)$ does not imply $y \in X$, then every set is fooling for α. As a consequence: P is fooling for a set Φ of formulas iff P is fooling for $\{\varphi \in \Phi \mid \varphi \rightarrow y \in X\}$. Theorem 5.1 and Corollary 5.2 imply that every formula has a fooling set and a fooling antichain.

▶ **Lemma 5.4.** *The following statements hold in the full binary tree:*

1. *There is no MSO-definable choice function.*
2. *Every $\alpha(X, y)$ has a fooling set.*
3. *Every finite set of formulas has a fooling set.*
4. *There is no MSO-definable choice function on antichains.*
5. *Every formula has a fooling antichain.*
6. *Every finite set of formulas has a fooling antichain.*

Proof. First we prove that implications (1) ⇒ (2) ⇒ (3) ⇒(1) hold in every structure. These and Theorem 5.1 imply that (1)-(3) are true in the full binary tree.

(1)⇒(2). If $\alpha(X, y)$ does not define a choice function and $\alpha(X, y) \rightarrow y \in X$, then there is a fooling set for α.

(2)⇒(3). Let $\alpha_1(X, y), \dots, \alpha_n(X, y)$ be a finite sequence of formulas such that $\alpha_i(X, y) \rightarrow y \in X$.

Define

$$Fool_i(X) := X \neq \emptyset \wedge \neg\exists! y\alpha_i(X, y)$$

$Fool_i(X)$ holds iff X is a fooling set for α_i.

If $\bigwedge_{i=1}^{n} Fool_i(X)$ is satisfiable, then there is a fooling set for $\{\alpha_i \mid i = 1, \dots n\}$.

Toward a contradiction we assume that $\bigwedge_{i=1}^{n} Fool_i(X)$ is unsatisfiable, and construct a formula that has no fooling set.

For $I \subseteq \{1,\dots,n\}$ define $FOOL_I(X) := \bigwedge_{i\in I} Fool_i(X) \wedge \bigwedge_{j\notin I} \neg Fool_j(X)$.

Observe that $FOOL_I(X)$ for $I \subsetneq \{1,\dots,n\}$ defines a partition, i.e., if $I_1 \neq I_2$, then $FOOL_{I_1}(X) \wedge FOOL_{I_2}(X)$ is unsatisfiable, and $\bigvee_{I\subsetneq\{1,\dots,n\}} FOOL_I(X)$ holds for every X.

Let c be a function which assigns to every proper subset of $\{1,\dots,n\}$ an element in its complement. For $I \subsetneq \{1,\dots,n\}$ define

$$\beta_I(X,y) := FOOL_I(X) \wedge \alpha_j(X,y) \text{ for } j = c(I) \notin I$$

and

$$\gamma(X,y) := \bigvee_{I\subsetneq\{1,\dots,n\}} \beta_I$$

For every P there is a unique $I \subsetneq \{1,\dots,n\}$ such that $FOOL_I(P)$. Hence, by the definition of γ: $\gamma(P,y)$ iff $FOOL_I(P) \wedge \alpha_j(P,y)$ for $j = c(I)$. Now, $FOOL_I(P)$ implies $\neg Fool_j(P)$ for $j = c(I)$. Therefore, if $P \neq \emptyset$ then $\exists! y \alpha_j(P,y)$, and therefore, $\exists! y \beta_I(P,y)$, and $\exists! y \gamma(P,y)$. A contradiction to "γ has a fooling set."

Hence, there is a fooling set for $\{\alpha_i \mid i = 1,\dots n\}$.

(3)⇒(1) trivial.

(4) - this is Corollary 5.2.

Finally, the proof of the equivalences between (4),(5) and (6) is obtained by replacing "fooling set" by "fooling antichain" in the proof of the equivalences between (1),(2) and (3). ◀

Recall that there are finitely many (semantically distinct) formulas of qr $\leq n$ with free variables X and y. We say that P is n**-fooling** if it is fooling for the formulas of qr $\leq n$. Hence, by Lemma 5.4,

▶ **Corollary 5.5.** *For every n there is an n-fooling set.*

6 Uncountable Cross-section Theorem

In this section we prove Uncountable Cross-section Theorem which is stated in the Introduction. The idea is to force infinitely many choices from a fooling set. Since every choice leads to at least two possibilities, infinitely many choices lead to uncountable many possibilities. Below are details.

Consider a relation $R^\infty_{choice}(X,Y)$ defined as: "Y is a subset of X such that for every n: $r^n l{\uparrow} \cap Y$ is a one element set." It is clear that $R^\infty_{choice}(X,Y)$ is MSO definable.

Let P be a subset of $\{l,r\}^*$ and let $r^*lP := \{u \mid u = r^n lv \text{ for } v \in P \text{ and } n \geq 0\}$. Below "$\exists^{2^{\aleph_0}} Y$" stands for "there are at least $2^{\aleph_0}$ different Y."

▶ **Proposition 6.1.**

1. *Assume that $\alpha(X,Y)$ has* qr $\leq n$ *and P is an $n+1$-fooling set. If the cross-section of $\alpha(X,Y)$ at r^*lP is included in the cross-section of $R^\infty_{choice}(X,Y)$ at r^*lP, then $T, r^*lP \models \exists Y \alpha(X,Y) \rightarrow \exists^{2^{\aleph_0}} Y \alpha(X,Y)$.*
2. *The relation $R^\infty_{choice}(X,Y)$ contains no* MSO*-definable relation with the same domain and only cross-sections of cardinality less than $2^{\aleph_0}$.*

Proof. (2) immediately follows from (1).

(1) First, observe that if P is an $n+1$-fooling set, then no $\beta(X,Y)$ of qr $\leq n$ can choose a unique one element subset of P, i.e., $\neg\big(\exists! Y \beta(P,Y) \wedge \exists Y(\beta(X,Y) \wedge \exists y Y = \{y\})\big)$. Indeed, if β chooses a one element subset of P, then $\exists Y \beta(X,Y) \wedge y \in Y$ chooses a unique element from P. This contradicts that P is $n+1$-fooling.

Second, observe that (T, P) is isomorphic to $(T, r^*lP)_{\geq r^n l}$ for every n, and (T, r^*lP) is obtained from $(T, \emptyset)$ by grafting (T, P) at an antichain $\{r^i l \mid i \in \mathbb{N}\}$.

Let Q be such that $T, r^*lP, Q \models \alpha(X, Y)$. Since $\forall Y(\alpha(r^*lP, Y) \to R^{\infty}_{choice}(r^*lP, Y))$, there are $w_i \in P$ such that $(r^i l\!\uparrow \cap\, Q) = \{r^i l w_i\}$. Let $\tau_i(X, Y)$ be the n-type of the subtree rooted at $r^i l$ in (T, r^*lP, Q). Then $T, P, \{w_i\} \models \tau_i(X, Y)$. Since P is an $n+1$-fooling set, by the first observation above, there is $W'_i \neq \{w_i\}$ such that $T, P, W'_i \models \tau_i(X, Y)$.

Let $\pi_r := v_1 v_2 \ldots$ be the rightmost branch in the full binary tree T, i.e., $v_i = r^i$ for $i \in \mathbb{N}$. For $I \subseteq \mathbb{N}$ define $Q_I := \{r^i l w_i \mid i \in I\} \cup \bigcup_{i \notin I} r^i l W'_i$. Then the assumptions of Lemma 2.5 hold for $\pi := \pi_r$, $T^1 := (T, r^*lP, Q)$ and $T^2 := (T, r^*lP, Q_I)$. Therefore, by Lemma 2.5, $T, r^*lP, Q_I \models \alpha$ for every $I \subseteq \mathbb{N}$. Since $Q_I \neq Q_{I'}$ for $I \neq I'$, we obtain $T, r^*lP \models \exists^{2^{\aleph_0}} Y \alpha(X, Y)$. ◀

The next proposition describes a property that is more natural than R^{∞}_{choice}, and which also implies Uncountable Cross-section Theorem.

▶ **Proposition 6.2.** *The relation "Y is a branch such that $X \cap Y$ is infinite" contains no* MSO*-definable relation with the same domain and only cross-sections of cardinality $< 2^{\aleph_0}$.*

7 Conclusions and Further Results

Let us introduce some weaker variants of uniformization. Let $R^* \subseteq R$ be two relations with the same domain. If the cross-sections of a relation R^* are at most l for some $l \in \mathbb{N}$ (respectively, finite, countable), we say that R^* is an l-relation (respectively, a finitary relation, an $\aleph_0$-relation).

If R^* is an l-relation for some $l \in \mathbb{N}$ (respectively, a finitary or an $\aleph_0$-relation) we say that R^* is an l-uniformizer of R (respectively, finitary uniformizer, $\aleph_0$-uniformizer).

We say that ψ is an l-uniformizer of φ in a structure $\mathcal{M}$, if the relation definable by ψ in $\mathcal{M}$ is an l-uniformizer of the relation definable by φ in $\mathcal{M}$.

$\mathcal{M}$ has the l-uniformization property if for every MSO formula φ there is an MSO formula ψ that is an l-uniformizer of φ in $\mathcal{M}$. Finitary (and $\aleph_0$) uniformizers and the finitary (and $\aleph_0$) uniformization property are defined similarly.

For $l \in \mathbb{N}$, we say that the **uniformization rank** of R is l if R has an MSO-definable l-uniformizer and either $l = 1$ or R has no MSO-definable $(l-1)$-uniformizer. We say that the uniformization rank of R is finitary if R has an MSO-definable finitary uniformizer and has no MSO-definable l-uniformizer for $l \geq 1$. We say that the uniformization rank of R is $\aleph_0$ if R has an MSO-definable $\aleph_0$-uniformizer and has no finitary uniformizer. We say that the uniformization rank of R is $2^{\aleph_0}$ if R has no uniformizer with the cross-section of cardinality $< 2^{\aleph_0}$.

Our result can be restated as:

▶ **Corollary 7.1.** *The only ranks for* MSO*-definable relations in the full binary tree are one, $\aleph_0$ and $2^{\aleph_0}$.*

Indeed, the finite cross-section theorem implies that the rank is one or infinite. The relation $y \in X$ has rank $\aleph_0$. The relations $R^{\infty}_{choice}(X, Y)$ and "Y is a branch such that $X \cap Y$ is infinite" have rank $2^{\aleph_0}$. For the MSO-definable (with parameters) relation the continuum hypothesis holds [1]. Therefore, the infinite cross-sections of MSO-definable relations are either countable or have cardinality $2^{\aleph_0}$. Hence, Corollary 7.1 holds.

7.1 Ranks over Integers

In [4] the uniformization of MSO over the integers with the successor function was investigated. Consider, $\varphi_2(X, Y) := \forall t(Y(t) \leftrightarrow \neg Y(t+1))$. Note that φ_2 does not depend on X and there are exactly two sets Y_{even} := "the set of even integers" and Y_{odd} :="the set of odd integers" that satisfy φ_2. Note also that there is an order preserving automorphism of integers that maps Y_{even} onto Y_{odd}. Therefore, no MSO formula distinguishes between Y_{even} and Y_{odd}. Hence, the uniformization rank of φ_2 is two. Similarly, for every l one can define an MSO formula φ_l which has the uniformization rank l over the integers. It was proved in [4] that the uniformization rank of an MSO formula over integers is computable.

It is open whether the uniformization rank of an MSO formula over the full binary tree is computable.

7.2 Ranks over Ordinals

Lifsches and Shelah [7] considered the uniformization problem over the class of trees and the class of ordinals. Recall that a structure $\mathcal{M}$ is said to have the uniformization property if every MSO definable relation (in $\mathcal{M}$) has an MSO definable uniformizer. Lifsches and Shelah proved that an ordinal α has the uniformization property iff $\alpha < \omega^\omega$.

In the full paper we consider uniformization over ordinals. We have not provided an algorithm to decide the uniformization rank of a formula. However, we prove that if an MSO definable relation over an ordinal has only countable cross-sections, then it has a uniformizer. Moreover, the only ranks for the MSO-definable relations over a countable ordinal are one and $2^{\aleph_0}$.

7.3 Uniformization Degrees

We know that the formula $\psi_1 := y \in X$ has no MSO uniformizer in the full binary tree. Now, let us look at the formula ψ_2 stating that"Y is a branch such that $Y \cap X$ is infinite."

This formula has no MSO uniformizer. But are there any other interesting relations between these two formulas? Can we say, for instance, that ψ_2 is even "harder" to uniformize than ψ_1 (whatever this might mean)? Or, perhaps the other way round? Do we feel that the example of ψ_2 "contains a new idea" when it comes to our discussion of uniformization? To turn these admittedly vague questions into mathematical ones, we require a notion of comparing formulas and perhaps an equivalence relation on them. However, as our example shows, the semantical equivalence seems not to be the right notion. Note, however, the following. For any set X let $X\uparrow := \{z \mid \exists x \in X(z \geq x)\}$ be the upward closure of X. X is non-empty iff there is a branch Y such that $|Y \cap X\uparrow|$ is infinite. Moreover, if $Y \cap X \neq \emptyset$, then $Y \cap X\uparrow$ has a minimal point y which is in X. Hence, by using any uniformizer for ψ_2 we succeeded to define (in MSO) a uniformizer for ψ_1.

This suggests the following definition.

Let $\mathcal{M}$ be a structure in a signature Σ and $\psi(\overline{X}, \overline{Y})$ and $\psi_2(\overline{U}, \overline{V})$ be formulas in Σ. ψ_1 is easier to uniformize (in $\mathcal{M}$) than ψ_2 if there is an MSO formula $\varphi(\overline{X}, \overline{Y})$ (reduction formula) in an expansion of Σ by a relational symbol c such that:

if $\mathcal{M}[c]$ is an expansion of $\mathcal{M}$, where c is interpreted as a uniformizer of ψ_2, then $\varphi(\overline{X}, \overline{Y})$ is a uniformizer of ψ_1.

We write $\psi_1 \preceq_{un} \psi_2$ if ψ_1 is easier to uniformize than ψ_2. The relation $\preceq_{un}$ is a preorder relation and its equivalence classes can be called (uniformization) degrees.

We can show that "$y \in X$," $R^{\infty}_{choice}(X, Y)$, "$Y$ is a branch such that $Y \cap X$ is infinite," "Y is a branch such that $Y \cap X$ is finite" and "Y is a finite non-empty subset of an antichain X" have the same degree. It is interesting to study the structure of degrees. In particular, we do not know whether there is a maximal degree.

References

1 Vince Bárány, Łukasz Kaiser, and Alex Rabinovich. Expressing cardinality quantifiers in monadic second-order logic over trees. *Fundamenta Informaticae*, 100(1-4):1–17, 2010.

2 Arnaud Carayol and Christof Löding. MSO on the infinite binary tree: Choice and order. In *International Workshop on Computer Science Logic*, pages 161–176. Springer, 2007.

3 Arnaud Carayol, Christof Löding, Damian Niwinski, and Igor Walukiewicz. Choice functions and well-orderings over the infinite binary tree. *Open Mathematics*, 8(4):662–682, 2010.

4 Grzegorz Fabianski, Michal Skrzypczak, and Szymon Torunczyk. Uniformisations of regular relations over bi-infinite words. In Holger Hermanns, Lijun Zhang, Naoki Kobayashi, and Dale Miller, editors, *LICS '20: 35th Annual ACM/IEEE Symposium on Logic in Computer Science, Saarbrücken, Germany, July 8-11, 2020*, pages 384–396. ACM, 2020. doi:10.1145/3373718.3394782.

5 Yuri Gurevich. Monadic second-order theories. *Model-theoretic logics*, pages 479–506, 1985.

6 Yuri Gurevich and Saharon Shelah. Rabin's uniformization problem 1. *The Journal of Symbolic Logic*, 48(4):1105–1119, 1983.

7 Shmuel Lifsches and Saharon Shelah. Uniformization and skolem functions in the class of trees. *The Journal of Symbolic Logic*, 63(1):103–127, 1998.

8 Michael O Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the american Mathematical Society*, 141:1–35, 1969.

9 Saharon Shelah. The monadic theory of order. *Annals of Mathematics*, 102(3):379–419, 1975.

10 Wolfgang Thomas. Automata on infinite objects. In *Formal Models and Semantics*, pages 133–191. Elsevier, 1990.

11 Wolfgang Thomas. Ehrenfeucht games, the composition method, and the monadic theory of ordinal words. In *Structures in Logic and Computer Science*, pages 118–143. Springer, 1997.

12 Boris Trakhtenbrot and Ya Martynovich Barzdin. *Finite automate: behaviour and synthesis*. North-Holland, 1973.