Regular Choice Functions and Uniformisations For countable Domains

Vincent Michielini
University of Warsaw, Poland
michielini@mimuw.edu.pl

Michał Skrzypczak
University of Warsaw, Poland
mskrzypczak@mimuw.edu.pl

Abstract

We view languages of words over a product alphabet $A \times B$ as relations between words over A and words over B. This leads to the notion of regular relations – relations given by a regular language. We ask when it is possible to find regular uniformisations of regular relations. The answer depends on the structure or shape of the underlying model: it is true e.g. for ω-words, while false for words over $\mathbb{Z}$ or for infinite trees.

In this paper we focus on countable orders. Our main result characterises, which countable linear orders D have the property that every regular relation between words over D has a regular uniformisation. As it turns out, the only obstacle for uniformisability is the one displayed in the case of $\mathbb{Z}$ – non-trivial automorphisms of the given structure. Thus, we show that either all regular relations over D have regular uniformisations, or there is a non-trivial automorphism of D and even the simple relation of choice cannot be uniformised. Moreover, this dichotomy is effective.

2012 ACM Subject Classification Theory of computation → Regular languages; Theory of computation → Automata over infinite objects

Keywords and phrases Uniformisation, Monadic Second-order logic, Countable words

Digital Object Identifier 10.4230/LIPIcs.MFCS.2020.69

Funding *Vincent Michielini*: The first author has been supported by the European Research Council (ERC) grant under the European Union's Horizon 2020 research and innovation programme (ERC Consolidator Grant LIPA, grant agreement No. 683080).
Michał Skrzypczak: The second author has been supported by Poland's National Science Centre (NCN) (grant No. 2016/21/D/ST6/00491).

1 Introduction

There are many ways of interpreting the simple mathematical operation of projection $\Pi_X : X \times Y \to X$. From the computer scientist's perspective, we often use the intuition of *guessing* that leads to the notion of non-determinism: the projection $\Pi_X(R)$ of a relation $R \subseteq X \times Y$ is the set of the elements $x \in X$ which admit at least one *witness* $y \in Y$ such that $(x, y) \in R$. In many cases this operation greatly increases the expressive power of the considered machines (e.g. in the case of recursively enumerable sets), while in other cases it does not (e.g. in the case of the class PSPACE). Also, the famous $\mathrm{P} \stackrel{?}{=} \mathrm{NP}$ problem asks about the strength of projection.

One of the ways of dealing with the complexity of that operation is to provide a constructive way of finding the witnesses y. This concept is formalised by the notion of a uniformisation: $F \subseteq R$ is a *uniformisation* of R if $\Pi_X(F) = \Pi_X(R)$ and for each $x \in \Pi_X(F)$ there is a **unique** $y \in Y$ such that $(x, y) \in F$ – thus, F is the graph of a partial function. It is known that in certain cases, if a relation admits a *definable* uniformisation then its projection is also *definable* (e.g. when *definable* = Borel). This is one of the many reasons motivating the question of uniformisation: which *definable* relations admit *definable* uniformisations?

In this paper we work with the automata-theoretic notion of *definability* i.e. definability in Monadic Second-Order logic (**MSO**) or equivalently: being a regular language. To speak about relations between structures over two alphabets A and B; we encode them as languages over the product alphabet $A \times B$. In this context, the coarsest question of uniformisation is well-understood: all regular relations admit regular uniformisations in the cases of finite and infinite words as well as finite trees [10, 7, 9]; while the celebrated result of Gurevich and Shelah [6, 1] shows that there are some regular relations over infinite trees that have no regular uniformisation. From this perspective, the case of countable linear orders seems to be simple, because already over bi-infinite words (words over $\mathbb{Z}$) the relation "choose a single position" has no regular uniformisation.

While some regular relations over specific structures (e.g. infinite trees) do not have regular uniformisations, some others may have. Thus, when working with a specific relation (possibly coming from some specification) or a specific shape of structures (e.g. countable words of certain fixed domain), one would like to ask the question of uniformisation for this particular case.

The aim of this paper is to approach this more fine-grained question of uniformisation in one of the simplest non-trivial cases: given a representation of a countable linear order D, decide if all regular relations between words of that domain admit regular uniformisations. Thus, the answer for $D = \{0, \dots, 9\}$ or $D = \omega$ should be **YES**, while the answer for $D = \mathbb{Z}$ should be **NO**. Our hope is that understanding well the obstacles for uniformisability in this case will later be useful in understanding the case of infinite trees – one can easily interpret every countable linear order as a set of vertices in a tree.

Our main result states, that for *representable* domains D, the problem if all regular relations over D have regular uniformisations is decidable. As it turns out, this question is equivalent to the question whether there is a regular choice function over D, which in turn is equivalent to the fact that D has no non-trivial automorphisms. This implies that the only obstacle for uniformisability over countable domains is the one present in $\mathbb{Z}$ – automorphisms of the structure.

This work is a part of a bigger project aiming at the questions of uniformisation. In particular, the recent paper [4] provides an effective characterisation, that given a regular relation between bi-infinite words (i.e. words over $\mathbb{Z}$), decides if that particular relation has a regular uniformisation. In the present paper we answer a coarser question, asking about all relations over a specific domain. These questions do not seem to be inter-reducible.

2 Background knowledge

An *alphabet* A is a finite non-empty set, and a *domain* D is a totally ordered set. In this paper are of particular interest countable domains (in the sense finite or of the cardinality of the set $\natural$ of natural numbers). An element $x \in D$ is called a *position* of D. A subset $X \subseteq D$ is called *convex* if for every three positions $x < y < z$ of D, if $x, z \in X$ then also $y \in X$. Given two subsets $X, Y \subseteq D$, we write $X < Y$ if for every pair $x \in X$ and $y \in Y$ we have $x < y$. Notice that $X < Y$ implies that $X \cap Y = \emptyset$. If two sets X, Y are known to be disjoint, then we emphasise this fact by denoting their union as $X \sqcup Y$. Given two positions $x, z \in D$, by $[x, z]$ we denote the convex set $\{y \in D \mid x \leq y \leq z\}$.

A word w over some alphabet A (or, more generally, over a set) is a function from a domain, denoted $Dom(w)$, to A. For a position $x \in D$, the value $w(x) \in A$ is called the *label* of x. The set of words over A with a domain D is denoted A^D and the set of all words over A for all countable domains is denoted A°. A *language* over A is any subset of A° or any subset of A^D for a fixed domain D. Given a word $w \in A^D$ and a non-empty convex subset $X \subseteq D$, by $w{\restriction}_X \in A^X$ we denote the restriction of w to the domain X. Moreover, we will sometimes work with the singleton alphabet $\{\centerdot\}$ and identify any word $w \in \{\centerdot\}^\circ$ with its domain $D = \mathit{Dom}(w)$.

To deal with alphabets which are the products of two sets, we use the following special notation: if $a \in A$ and $b \in B$, then $\binom{a}{b}$ is the product letter in $A \times B$; and if w, σ are words over the same domain D and over A and B respectively, then $\binom{w}{\sigma}$ denotes the word in $(A \times B)^D$ such that for all $s \in D$, $\binom{w}{\sigma}(s) = \binom{w(s)}{\sigma(s)}$.

Let D_1 and D_2 be two domains, an *isomorphism* from D_1 to D_2 (or between D_1 and D_2) is a bijective function ι which preserves the order, meaning that for all $x < y \in D_1$, $\iota(x) < \iota(y)$. If w_1 and w_2 are two words over A, then an isomorphism from w_1 to w_2 (or between w_1 and w_2) is an isomorphism ι from $\mathit{Dom}(w_1)$ to $\mathit{Dom}(w_2)$ which additionally preserves the labels: for all $x \in \mathit{Dom}(w_1)$, $w_1(x) = w_2(\iota(x))$. Two words or domains are said *isomorphic* to each other if there exists an isomorphism between them. Isomorphic words and domains will be sometimes identified in this paper. An *automorphism* of a word w (resp. of a domain D) is an isomorphism from w (resp. D) to itself. An automorphism is called *non-trivial* if it is not the identity function.

A word whose domain is finite is called a *finite word*. The set of all finite non-empty words over A is denoted A^+ and $A^* \stackrel{\text{def}}{=} A^+ \cup \{\epsilon\}$ contains additionally the empty word ϵ. A word whose domain is isomorphic to the set $\omega = \{0, 1, 2 \dots\}$ of natural numbers is called an *ω-word*. Another important domain in the paper is the set $\omega^\star = \{\dots, -3, -2, -1\}$.

Up to isomorphism, there exists a unique word w over A whose domain is countable and without borders (i.e. without maximal nor minimal elements), and which is densely labelled in the following sense: for all $x < z \in \mathit{Dom}(w)$ and $a \in A$, there exists $y \in \mathit{Dom}(w)$ such that $x < y < z$ and $w(y) = a$. We call this word the *perfect shuffle* of A, and denote it A^η. We often identify $\mathit{Dom}(A^\eta)$ with $\mathbb{Q}$, $\mathbb{Q}$ being, up to isomorphism, the only countable and dense domain without borders.

If $(w_i)_{i \in I}$ is an indexed family of words, I itself being a domain, then by $\sum_{i \in I} w_i$ we denote the *concatenation* of the w_i's, defined as being the word w of domain $\bigsqcup_{i \in I}\{\langle i, x_i\rangle \mid x_i \in \mathit{Dom}(w_i)\}$, defined by $w(\langle i, x_i\rangle) = w_i(x_i)$ for each $i \in I$ and $x_i \in \mathit{Dom}(w_i)$. The domain $\bigsqcup_{i \in I}\{\langle i, x_i\rangle \mid x_i \in \mathit{Dom}(w_i)\}$ is totally ordered by $\langle i, x_i\rangle \leq \langle j, y_j\rangle$ if $i < j$, or $i = j$ and $x_i \leq y_i$ in $\mathit{Dom}(w_i)$.

We have special notations for some particular cases: $w_0 \cdot w_1$ if $I = \{0, 1\}$, and w^ω (resp. $w^{\omega^\star}$) if $I = \omega$ (resp. $\omega^\star$) and all the w_i's are isomorphic to w. We write $w^\mathbb{Z}$ for $w^{\omega^\star} \cdot w^\omega$. Similarly, we write w^n in the case $I = \{0, \dots, n{-}1\}$ and all the w_i's are isomorphic to w. Finally, if $w_0, \dots, w_{n-1}$ are words over A then $\{w_i \mid i \in n\}^\eta$ denotes the word $\sum_{q \in \mathbb{Q}} w_{u(q)}$, where $u = \{0, \dots, n-1\}^\eta$, obtained as the *perfect shuffle* of the words w_i.

A word $w \in A^\circ$ is called *finitary* (some literature also uses the term *regular*) if it can be constructed from single letters using a finite number of applications of the operations $\cdot$, $(.)^\omega$, $(.)^{\omega^\star}$, and $(.)^\eta$, see Section 4. It is easy to see that only countably many words are finitary. As we identify words over the single-letter alphabet $\{\centerdot\}$ with their domains, it also makes sense to say that a domain is *finitary*. Notice that a non-finitary word may however have a finitary domain: it is for example the case of the non-finitary word $\sum_{i \in \omega} a^i b$, whose domain is ω. An example of a non-finitary domain is the countable ordinal ω^ω, where here we treat the operation $(.)^\omega$ in the ordinal-theoretic sense.

○-semigroups

Similarly as semigroups provide an algebraic framework to recognise regular languages of finite words [8], ○-semigroups [2] allow to recognise languages of countable words. A *○-semigroup* is a pair $\langle S, \pi\rangle$ where S is a non-empty set and π is a function from S° to S, which satisfies the following property of *generalised associativity*: for every family of words $(w_i)_{i\in I} \subseteq S^\circ$, indexed by a countable domain I, we have

$$\pi\left(\sum_{i\in I} \pi(w_i)\right) = \pi\left(\sum_{i\in I} w_i\right), \tag{1}$$

where the left-hand side sum ranges over single letter words $\pi(w_i)$; and the right-hand side sum is just the concatenation of all the words w_i. We often identify a ○-semigroup $\langle S, \pi\rangle$ with its set S.

To make a representation of a ○-semigroup finite, one uses a concept of a *○-algebra* – a quintuple $\langle S, \cdot, (.)^\tau, (.)^{\tau^\star}, (.)^\kappa\rangle$, where $\langle S, \cdot\rangle$ is a semigroup, $(.)^\tau$ and $(.)^{\tau^\star}$ are unary operations over S, and $(.)^\kappa\colon \mathcal{P}^{\mathrm{fin}}_+(S) \to S$ is called a *shuffle* operation, that assigns elements of S to all finite non-empty subsets of S. We additionally require the above operations to satisfy certain axioms, see [2, Definition 2]. Again, we often identity the ○-algebra with the set S itself.

Each ○-semigroup induces a ○-algebra by defining $s\cdot t = \pi(st)$, $s^\tau = \pi(s^\omega)$, $s^{\tau^\star} = \pi(s^{\omega^\star})$, and $P^\kappa = \pi(P^\eta)$, where s is treated as a single-letter word and st is a two-letter word. One of the main results of [2], Theorem 11, states that every finite ○-algebra is induced by a unique ○-semigroup – in other words, there is a unique way to define a product operation $\pi\colon S^\circ \to S$ in a way satisfying (1) that is additionally consistent with the above equations.

Notice that the operation $\pi_\Sigma\big((w_i)_{i\in I}\big) \overset{\mathrm{def}}{=} \sum_{i\in I} w_i$ itself satisfies (1), and therefore $\langle A^\circ, \pi_\Sigma\rangle$ is a ○-semigroup, which is called the *free ○-semigroup* on A. It induces the *free ○-algebra* $\langle A^\circ, \cdot, (.)^\omega, (.)^{\omega^\star}, (.)^\eta\rangle$.

A *homomorphism* is a function between two algebraic structures that preserves all their operations. We say that a language L of countable words over A is *recognised* by a ○-semigroup $\langle S, \pi\rangle$ if there exists a homomorphism h from $\langle A^\circ, \pi_\Sigma\rangle$ to $\langle S, \pi\rangle$ such that $L = h^{-1}(H)$ for some $H \subseteq S$ (or equivalently such that $L = h^{-1}(h(L))$).

A language $L \subseteq A^\circ$ is *regular* if it is recognised by some finite ○-semigroup. For a fixed domain D, a language $L \subseteq A^D$ is called *regular over the domain D* if $L = A^D \cap L'$ for some regular language $L' \subseteq A^\circ$.

The following fact is an important consequence of the correspondence between ○-semigroups and ○-algebras. It implies that finitary words are distinctive for regular languages.

▶ **Proposition 1** ([2, Theorem 13]). *If $L \neq \emptyset$ is regular then L contains a finitary word.*

Monadic Second Order Logic

One of the classical ways of characterising general regular languages is expressed in terms of logical definability. In this exposition we follow the ideas and notation from [5, Section 12]. Monadic Second-Order logic (**MSO**) is an extension of First-Order logic [3] by additional *monadic* quantifiers $\exists X.\ \psi(X)$ and $\forall X.\ \psi(X)$ that range over subsets of the domain. In this work we are interested in words, treated as logical structures. Thus, given a word $w \in A^\circ$ with some domain $D = \mathit{Dom}(w)$, we treat it as a relational structure with universe D, binary relation $\leq$ representing the order on D, and unary predicates $a \in A$, such that $a(x)$ if and only if $w(x) = a$. This way it makes sense to ask if a given **MSO** sentence φ *holds* or is *satisfied* over a word w. The *language* of a formula φ over an alphabet A, denoted $\mathcal{L}(\varphi) \subseteq A^\circ$, is the set of all words satisfying φ.

One can easily encode a formula $\varphi(X_0, \dots, X_{n-1})$ over an alphabet A with free variables $X_0, \dots, X_{n-1}$ as a sentence φ over the alphabet $A \times \{0,1\}^n$, whose symbols should be seen as characteristic functions of the parameters $X_0, \dots, X_{n-1}$ (we can treat each first-order variable as a second-order variable evaluated in a singleton set).

▶ Remark 2. If w_1 and w_2 are two isomorphic words and φ is an **MSO**-sentence, then $w_1 \in \mathcal{L}(\varphi)$ if and only if $w_2 \in \mathcal{L}(\varphi)$.

▶ **Theorem 3** ([2, Theorems 28 and 31]). *A language $L \subseteq A^\circ$ is regular if and only if there exists an **MSO**-sentence φ such that $\mathcal{L}(\varphi) = L$. Moreover, there exist effective translations between: a finite $\circ$-algebra recognising L and an **MSO**-sentence whose language is L.*

Uniformisation and choice

Given two sets X and Y, a relation $R \subseteq X \times Y$ is *functional* if for every x in the projection $\Pi_X(R)$ of R onto X, there exists a unique $y \in Y$ such that $(x, y) \in R$. We say that $F \subseteq X \times Y$ is a *uniformisation* of $R \subseteq X \times Y$ if $F \subseteq R$; $\Pi_X(F) = \Pi_X(R)$; and F is functional. Thus, a uniformisation is a way of choosing a single witness $y \in Y$ for each $x \in \Pi_X(R)$ in such a way that $(x, y) \in R$.

Fix two alphabets A and B. We say that a relation $R \subseteq A^\circ \times B^\circ$ is *synchronised* if for each $(w, \sigma) \in R$ we have $Dom(w) = Dom(\sigma)$. Each synchronised relation R can be identified with a language $L_R = \{\binom{w}{\sigma} \mid (w, \sigma) \in R\} \subseteq (A \times B)^\circ$ over the product alphabet $A \times B$. A synchronised relation is *regular* if so is the language L_R. Analogously, a relation $R \subseteq A^D \times B^D$ is *regular* over a domain D if L_R is a regular language over D.

The crucial question of this paper asks, which regular relations $R \subseteq A^\circ \times B^\circ$ admit uniformisations $F \subseteq R$ which are also regular. In other words, we seek for a regular (or **MSO**-definable) way to pick, for each word $w \in \Pi_{A^\circ}(R)$, a single word $\sigma \in B^{Dom(w)}$ such that $(w, \sigma) \in R$.

One of the simplest instances of the uniformisation question is the one when R is the *membership relation*: both alphabets A and B are $\{0,1\}$, and the relation R requires that the letter $\binom{1}{1}$ appears exactly once, while the letter $\binom{0}{1}$ does not appear at all. In other words, R corresponds to the language $L_R = \mathcal{L}(\varphi_{\text{member}}) \subseteq \left(\{0,1\}^2\right)^\circ$ of the formula $\varphi_{\text{member}}(X, y) \equiv y \in X$. To find a regular uniformisation of R boils down to define a *regular choice function*: a regular relation that selects a single element y from every non-empty set $X \subseteq Dom(w)$ of positions of a given word w.

Classical results [10, 7, 9] show that regular relations always admit regular uniformisations in the following two cases.

▶ **Theorem 4.** *Every regular relation between finite words $R \subseteq A^+ \times B^+$, or ω-words $R \subseteq A^\omega \times B^\omega$ effectively admits a regular uniformisation.*

However, over the domain $\mathbb{Z}$ there does not even exist any regular choice function. Indeed, the domain admits automorphisms $y \mapsto y+n$ for each $n \in \mathbb{Z}$, and therefore all the positions *look the same* and we cannot define in a regular way a unique position for the full domain $\mathbb{Z}$.

The above observations motivate the following question: given a domain D, decide if all regular relations over the domain D admit regular uniformisations over D. If it is the case then we say that D has the *regular uniformisation property*, or, more simply, the *uniformisation property*.

3 Main result

The main result of this work provides an effective characterisation for the question when a given finitary domain D has the uniformisation property.

▶ **Theorem 5.** *Let D be a finitary domain. The following conditions are equivalent:*

i) *D admits a regular choice function;*
ii) *D has the uniformisation property;*
iii) *D does not admit a non-trivial automorphism;*
iv) *D does not have any convex subset isomorphic to $I^{\mathbb{Z}}$, i.e. $\mathbb{Z}$ consecutive copies of I, generally denoted $I \times \mathbb{Z}$ in the literature, for any non-empty domain I.*

*Moreover, Items i) and ii) are effective: given a representation of D one can either compute a choice function and a procedure for constructing regular uniformisations; or return **NO** meaning that the above conditions fail for D.*

The above statement is expressed in terms of a given finitary domain D and relations over it. However, the presented techniques apply equally well to a given finitary word $w \in A^{\circ}$ and regular relations $R \subseteq B^D \times C^D$ definable over w – such a relation is given by a regular language L_R over the domain D and the alphabet $A \times B \times C$, by $R = \{(u, \sigma) \in B^{Dom(w)} \times C^{Dom(w)} \mid \left(\begin{smallmatrix} w \\ u \\ \sigma \end{smallmatrix}\right) \in L_R\}$. In that case, the regular relations over the word $w = a^{\omega^\star} \cdot b^{\omega}$ *do* admit regular uniformisations, because w does not have any non-trivial automorphism. On the other hand, the word $w = (ab)^{\mathbb{Z}}$ from Figure 1 below admits many non-trivial automorphisms and therefore violates the above conditions. For the sake of notational simplicity, most of the proof is given in terms of domains D, i.e. words over $\{\centerdot\}$.

We would like to emphasise that the above result does not hold for non-finitary finitary domains. A counterexample is the domain $D = \omega^{\omega}$ (again $(.)^{\omega}$ here is treated in the ordinal-theoretic sense): it is an ordinal and therefore satisfies Items i, iii, and iv, but it does not have the regular uniformisation property, as it was proved by Lifsches and Shelah in [7].

Certain implications of the above theorem are straightforward. A regular choice function is a special case of a uniformisation question, so Item ii) implies Item i). Also, Items iii) and iv) are easily equivalent, because if $\iota\colon D \to D$ is an automorphism such that $\iota(x_0) \neq x_0$ then the set $\{\iota^k(x_0) \mid k \in \mathbb{Z}\}$ is order-isomorphic to $\mathbb{Z}$. Moreover, any non-trivial automorphism can be used to disprove the existence of a regular choice function, so Item i) implies iii). Therefore, the only missing part of the proof is the implication $iii) \Rightarrow ii)$ and the effectiveness of these constructions.

The following remark follows from the fact that for every finite set A, the word A^{η} is isomorphic to $\left(A^{\eta}\right)^{\mathbb{Z}}$. In the particular case of A being the singleton alphabet $\{\centerdot\}$, it boils down to the fact that $\mathbb{Q}$ is isomorphic to $\mathbb{Q} \times \mathbb{Z}$, i.e. $\mathbb{Z}$ copies of $\mathbb{Q}$.

▶ Remark 6. If the construction of D in the $\circ$-algebra $\langle \{\centerdot\}^{\circ}, \cdot, (.)^{\omega}, (.)^{\omega^\star}, (.)^{\eta} \rangle$ involves any application of the operation $(.)^{\eta}$ then necessarily D does not satisfy Item iv).

Therefore, for the rest of the construction we can assume that D is *scattered*, i.e. it is constructed from the symbol $\centerdot$ using only the operations $\cdot$, $(.)^{\omega}$, and $(.)^{\omega^\star}$ in $\{\centerdot\}^{\circ}$.

The proof of the implication $iii) \Rightarrow ii)$ is based on a concept of *tree decompositions* of D. Such a *tree decomposition* is an **MSO**-definable object that represents a possible way how to obtain D as an evaluation of a fixed term in $\langle \{\centerdot\}^{\circ}, \cdot, (.)^{\omega}, (.)^{\omega^\star} \rangle$. Proposition 8 shows that there is a bijection between tree decompositions of D and automorphisms of D. Therefore, under the assumption of Item iii), there is a unique tree decomposition of D that corresponds to the identity automorphism of D. Based on that decomposition, one can effectively construct regular uniformisation of any given regular relation over the domain D.

Additionally, due to **MSO** definability of tree decompositions (see Proposition 10 below), there exists a fixed **MSO** sentence ψ_{unique} that expresses that a given domain D admits exactly one tree decomposition. Therefore, Item iii) holds if and only if D satisfies ψ_{unique}, which can be effectively checked.

4 Trees and terms

This section introduces the concepts of ranked trees that represent the way how a finitary scattered word $w \in A^\circ$ is obtained from single letters via the operations $\cdot$, $(.)^\omega$, and $(.)^{\omega^\star}$. These concepts are later used to define tree decompositions.

A *ranked set* is a finite set of *ranked symbols*, where each *ranked symbol* ℓ has its *arity* $\text{ar}(\ell) \subseteq \mathbb{Z}$ – a (possibly empty) convex set of integers. If $\text{ar}(\ell) = \emptyset$ then we call ℓ *nullary*; if $\text{ar}(\ell) = \{0\}$ then ℓ is *unary*; and if $\text{ar}(\ell) = \{0, 1\}$ then ℓ is *binary*.

A *ranked tree* over a fixed ranked set is defined inductively: if ℓ is a ranked symbol and $(t_i)_{i \in I}$ for $I = \text{ar}(\ell)$ is a family of ranked trees indexed by the arity of ℓ then there exists a ranked tree that is denoted $\ell[(t_i)_{i \in I}]$. We use the following notations for the tree $\ell[(t_i)_{i \in \text{ar}(\ell)}]$: $\ell[\,]$ when ℓ is nullary; $\ell[t_0]$ when ℓ is unary; and $\ell[t_0, t_1]$ when ℓ is binary.

Each ranked tree $t = \ell[(t_i)_{i \in I}]$ can be seen as a structure consisting of the set of *nodes* $\text{nodes}(t)$ (formally elements of $\mathbb{Z}^*$ – finite sequences of integers), defined inductively: $\text{nodes}(t) = \{\epsilon\} \cup \bigcup_{i \in I}\{iv \mid v \in \text{nodes}(t_i)\}$. The node $v = \epsilon$ is called the *root* of t; the nodes iv for $i \in I$ are called *children* of v; and v is the *father* of each of its children iv. A *leaf* is a node that has no children – it must be labelled by a nullary symbol. By $\text{leafs}(t)$ we denote the set of all leafs of t.

Each node v of t *indicates* a subtree of t: ϵ indicates t and a node of the form iv indicates the subtree of t_i indicated by v. The transitive reflexive closure of the father-child relation is the prefix order $\preceq$ on $\text{nodes}(t) \subseteq \mathbb{Z}^*$. Additionally, the set of nodes of t is ordered by the lexicographic order $\leq_{\text{lex}}$ in $\mathbb{Z}^*$.

We will work with two ranked sets for each fixed alphabet A. The first, corresponds to the operations of a $\circ$-algebra: $A \sqcup \{(\cdot), (\times\omega), (\times\omega^\star)\}$, where each symbol $a \in A$ is nullary, $(\cdot)$ is binary, and $(\times\omega)$, $(\times\omega^\star)$ are unary. A ranked tree over this ranked set is called a *term*. Notice that the arities of this ranked set are finite and therefore each term is a finite object.

Our second ranked set represents actual decompositions of a given countable word over an alphabet A. Its symbols are $A \sqcup \{(+), (\Sigma_\omega), (\Sigma_{\omega^\star})\}$, where again each symbol $a \in A$ is nullary, $(+)$ is binary, $\text{ar}((\Sigma_\omega)) = \omega$, and $\text{ar}((\Sigma_{\omega^\star})) = \omega^\star$ – the arity of the last two symbols is infinite. A ranked tree over this ranked set is called a *condensation tree* (see [2, Definition 7]).

The operations of a $\circ$-algebra provide a natural way of obtaining a condensation tree (denoted $\text{tree}(\tau)$) from a term τ, that is defined inductively: $\text{tree}\big(a[\,]\big)$ is $a[\,]$ (for $a \in A$); $\text{tree}\big((\cdot)[\tau_0, \tau_1]\big)$ is $(+)[\text{tree}(\tau_0), \text{tree}(\tau_1)]$; $\text{tree}\big((\times\omega)[\tau_0]\big)$ is $(\Sigma_\omega)[(\text{tree}(\tau_0))_{i \in \omega}]$; and $\text{tree}\big((\times\omega^\star)[\tau_0]\big)$ is $(\Sigma_{\omega^\star})[(\text{tree}(\tau_0))_{i \in \omega^\star}]$.

For an example of the above construction, see Figure 1. Notice that each node v of $\text{tree}(\tau)$ is *obtained* from a particular node of τ: the $a[\,]$ node is *obtained* from the respective $a[\,]$ node in τ, similarly $(+)$ is *obtained* from $(\cdot)$, (Σ_ω) from $(\times\omega)$, and $(\Sigma_{\omega^\star})$ from $(\times\omega^\star)$.

Given a condensation tree t, by $\text{word}(t)$ we denote the word whose domain is $\text{leafs}(t)$ ordered by $\leq_{\text{lex}}$ and labelled as follows: consider a position $v \in \text{leafs}(t)$ of $\text{word}(t)$, v has to indicate a subtree of t of the form $a[\,]$ with $a \in A$, then v is labelled by a in $\text{word}(t)$.

The above definitions are constructed in such a way, that for each term τ, the word w obtained by evaluating τ in the free $\circ$-algebra is isomorphic with the word $\text{word}(\text{tree}(\tau))$, which we simply write $\text{word}(\tau)$. This allows us to formally define *finitary* words as those of the form $\text{word}(\tau)$ for a term τ.

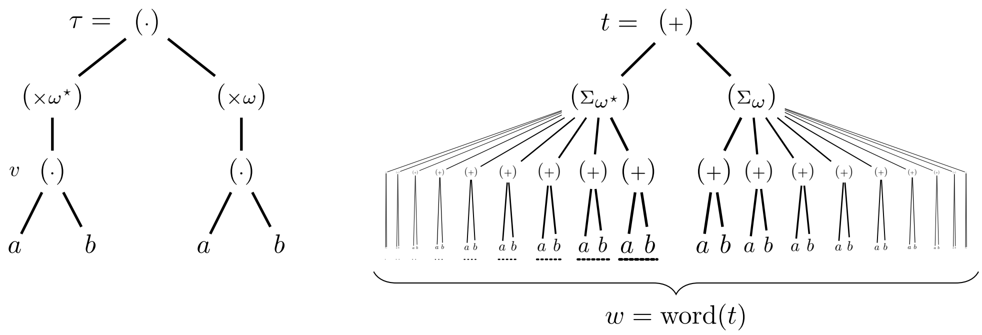

Figure 1 A term $\tau = (\cdot)\Big[(\times\omega^\star)\big[(\cdot)[a[\,], b[\,]\big], (\times\omega)\big[(\cdot)[a[\,], b[\,]\big]\Big]$, the tree $t = \mathrm{tree}(\tau)$, and the word $w = \mathrm{word}(t)$. Additionally, for v being the left $(\cdot)$ node of τ, the condensation C_v of w from the canonical tree decomposition Ξ_0 is marked by dashed intervals, its pieces are sub-words ab produced by the $(\times\omega^\star)$ sub-term.

▶ Remark 7. Given: a finitary word $w = \mathrm{word}(\tau)$ (represented as a term τ); a finite $\circ$-algebra S (represented explicitly by tables of its operations) and a homomorphism $h\colon A^\circ \to S$ (represented by the values $h(s) \in S$ for $a \in A$); one can effectively compute the value $h(w) \in S$. In particular, for every regular language $L \subseteq A^\circ$ (given either by a homomorphism to a finite $\circ$-algebra or by an **MSO** sentence and using [2, Theorem 27]), the membership problem $\mathrm{word}(\tau) \in L$ with input τ is decidable.

Tree decompositions

Fix a term τ and consider a word $w \in A^\circ$. In this section we define a concept of a *tree decomposition* with shape τ of w. Intuitively, such a tree decomposition (if it exists) provides a way of aligning w with $\mathrm{leafs}(\mathrm{tree}(\tau))$, i.e. encodes an isomorphism between w and $\mathrm{word}(\tau)$.

This construction follows some ideas from [2, Section 5], using the concept of *condensations*. A *condensation*[1] C on a word w is an equivalence relation on a non-empty subset of $Dom(w)$ (which is denoted $Dom(C)$) such that every equivalence class of C is a convex set, i.e. if $x < y < z$, x and z belong to $Dom(C)$, and $(x, z) \in C$ then y also belongs to $Dom(C)$ and $(x, y), (y, z) \in C$. An equivalence class K of C is called a *piece* of C.

A *tree decomposition* with shape τ is a family $\Xi = (C_v)_{v\in\mathrm{nodes}(\tau)}$ of condensations on w indexed by the nodes of τ, that additionally satisfies the following conditions. First, if v is a node of τ that is not a leaf and $(v_i)_{i\in I}$ are the children of v (in fact I equals $\{0\}$ or $\{0,1\}$) then

$$Dom(C_v) = \bigsqcup_{i\in I} Dom(C_{v_i}); \tag{2}$$

the union taken above must be disjoint; and for each $i \in I$ each piece of C_{v_i} must be contained in a single piece of C_v. Moreover, the following inductive conditions must hold.

1. If $v \in \mathrm{nodes}(\tau)$ is the root of τ then $Dom(C_v) = Dom(w)$ and C_v has a single piece consisting of the whole domain $Dom(w)$, i.e. $C_v = Dom(w)^2$ is the full relation.
2. If $v \in \mathrm{nodes}(\tau)$ is a binary node labelled by $(\cdot)$ with two children $v_0 \leq_{\mathrm{lex}} v_1$ then for every piece K of C_v we have that:

[1] For technical reasons we consider condensations with arbitrary domains – possibly different than the whole domain of a given word.

- for each $i \in \{0,1\}$, there is a single piece K_i of C_{v_i} that is contained in K,
- and $K_0 < K_1$ with $K_0 \sqcup K_1 = K$.

3. If $v \in \mathrm{nodes}(\tau)$ is a unary node labelled by $(\times\omega)$ with a single child v_0 then for every piece K of C_v we have that:
   - the set of pieces of C_{v_0} that are contained in K is of the form $\{K_n \mid n \in \mathbb{N}\}$, with
   - $K_0 < K_1 < K_2 < \ldots$ and $\bigsqcup_{n\in\mathbb{N}} K_n = K$.
4. If $v \in \mathrm{nodes}(\tau)$ is a unary node labelled by $(\times\omega^*)$ with a single child v_0 then for every piece K of C_v we have that:
   - the set of pieces of C_{v_0} that are contained in K is of the form $\{K_{-n} \mid n \in \mathbb{N} \setminus \{0\}\}$, with
   - $\cdots < K_{-3} < K_{-2} < K_{-1}$ and $\bigsqcup_{n\in\mathbb{N}\setminus\{0\}} K_{-n} = K$.
5. If $v \in \mathrm{nodes}(\tau)$ is a leaf of τ labelled by $a \in A$ then every piece of C_v must be a singleton $\{x\}$ such that $w(x) = a$.

Our aim now is the following proposition.

▶ **Proposition 8.** *Fix a term τ and a word $w \in A^\circ$. There exists a bijection $\Xi \mapsto \iota(\Xi)$ between tree decompositions Ξ with shape τ of w and isomorphisms $\iota(\Xi)\colon w \to \mathrm{word}(\tau)$.*

Before moving to its proof, we argue that tree decompositions with shape τ of a word w can be represented in **MSO** over w.

Representing tree decompositions in MSO

We begin by providing a representation in **MSO** over a word w of condensations C. First, if $X \subseteq D$ is any set, then it induces a symmetric relation $x \sim_X y$ on positions $x, y \in D$, such that for $x \leq y$ we have $x \sim_X y$ if $[x, y] \subseteq D$ and either $[x, y] \subseteq X$ or $[x, y] \cap X = \emptyset$. It is easy to check that for each set X, the above relation is a condensation, see [2, Lemma 34]. Now, a condensation C can be represented as a pair of sets (D, X) such that $D = \mathit{Dom}(C)$; $X \subseteq D$; and $x, y \in D$ are in the same piece of C if and only if $x \sim_X y$.

▶ **Lemma 9** ([2, Lemma 34]). *Every condensation C admits a representation (D, X) as above. Each pair (D, X) with $X \subseteq D \neq \emptyset$ represents some condensation.*

Notice that two pairs (D, X) and (D', X') represent the same condensation if and only if

$$D = D' \text{ and for every pair } x, y \in D \text{ we have } x \sim_X y \Leftrightarrow x \sim_{X'} y, \tag{3}$$

which provides an **MSO** definition of equality of condensations based on their representations.

▶ **Proposition 10.** *Take a term τ. There exists an **MSO** formula $\psi_{\mathrm{TD}(\tau)}\big((D_v, X_v)_{v\in\mathrm{nodes}(\tau)}\big)$ that holds over a word w and sets $\big(D_v, X_v\big)_{v\in\mathrm{nodes}(\tau)}$ if and only if for every $v \in \mathrm{nodes}(\tau)$ the pair (D_v, X_v) represents a condensation C_v and these condensations $(C_v)_{v\in\mathrm{nodes}(\tau)}$ form a tree decomposition with shape τ of w.*

The construction of this formula mostly follows literally the requirements above. Item 3 (and symmetrically Item 4) is expressed by guessing a set Y containing one element from each piece K_n and requiring that Y is of order type ω.

A condensation C of a word w is formally a subset of $\mathit{Dom}(w)^2$. This means that if $\iota\colon \mathit{Dom}(w) \to \mathit{Dom}(w')$ is an isomorphism between two words, then $\iota(C) \stackrel{\text{def}}{=} \{(\iota(x), \iota(y)) \mid (x, y) \in C\}$ is a condensation of w'. Moreover, if (D, X) represents C then $(\iota(D), \iota(X))$ represents $\iota(C)$. Therefore, Remark 2 and Proposition 10 imply the following corollary.

▶ **Corollary 11.** *If* $\iota\colon Dom(w) \to Dom(w')$ *is an isomorphism and* $\Xi = (C_v)_{v\in\mathrm{nodes}(\tau)}$ *is a tree decomposition with shape* τ *of* w *then* $\big(\iota(C_v)\big)_{v\in\mathrm{nodes}(\tau)}$ *is a tree decomposition with shape* τ *of* w'.

From tree decompositions to isomorphisms

We will now show how to define an isomorphism $\iota(\Xi)$ based on a tree decomposition Ξ.

▶ **Lemma 12.** *Let* $\Xi = (C_v)_{v\in\mathrm{nodes}(\tau)}$ *be a tree decomposition with shape* τ *of a word* w. *Consider a node* $v \in \mathrm{nodes}(\tau)$ *of* τ *that indicates a sub-term* τ'. *Let* K *be a piece of* C_v. *Then there exists an isomorphism* $\iota(\Xi)_{v,K}$ *between* $w{\restriction}_K$ *and* $\mathrm{word}(\tau')$.

This lemma is proved by induction. For v being a leaf of $\mathrm{tree}(\tau)$ each piece of C_v is a singleton, so the isomorphism is obvious. For other v one constructs $\iota(\Xi)_{v,K}$ by merging the isomorphisms $\iota(\Xi)_{v',K'}$ for v' being the children of v in $\mathrm{tree}(\tau)$. By $\iota(\Xi)$ we denote the above isomorphism for the root ϵ of τ, i.e. $\iota(\Xi) \stackrel{\mathrm{def}}{=} \iota(\Xi)_{\epsilon,Dom(w)}$.

▶ **Lemma 13.** *If* $\Xi = (C_v)_{v\in\mathrm{nodes}(\tau)}$ *and* $\Xi' = (C'_v)_{v\in\mathrm{nodes}(\tau)}$ *are two distinct tree decompositions of a word* w, *both with shape* τ, *then the isomorphisms* $\iota(\Xi)$ *and* $\iota(\Xi')$ *are distinct.*

This proof is a simple analysis of the definition of $\iota(\Xi)$.

From isomorphisms to tree decompositions

Now we provide the opposite transformation: from an isomorphism to a tree decomposition.

▶ **Lemma 14.** *There exists a* canonical *tree decomposition* Ξ_0 *with shape* τ *of the word* $\mathrm{word}(\tau)$. *Moreover,* $\iota(\Xi_0) = \mathrm{id}_{Dom(w)}$.

This tree decomposition is defined as follows. Take $v \in \mathrm{nodes}(\tau)$ and recall that each node of $\mathrm{tree}(\tau)$ is *obtained* from a unique node of τ, in the sense of the definition on page 7. For a pair of leaves x, y of $\mathrm{tree}(\tau)$ we let $(x, y) \in C_v$ if $u' \preceq x$ and $u' \preceq y$ for some $u' \in \mathrm{nodes}(\mathrm{tree}(\tau))$ that is obtained from v. It is easy to check that there is at most one such u' as above and C_v defined that way is in fact an equivalence relation and $\iota(\Xi_0) = \mathrm{id}_{Dom(w)}$.

▶ **Lemma 15.** *Fix a term* τ *and let* ι_0 *be an isomorphism between a word* $w \in A^\circ$ *and* $\mathrm{word}(\tau)$. *Then there exists a tree decomposition* Ξ *with shape* τ *of* w *such that* $\iota(\Xi) = \iota_0$.

Proof. Let $\Xi_0 = (C_v)_{v\in\mathrm{nodes}(\tau)}$ be the canonical tree decomposition of $\mathrm{word}(\tau)$. Define $\Xi = \big(\iota_0^{-1}(C_v)\big)_{v\in\mathrm{nodes}(\tau)}$. By Corollary 11 we know that Ξ is a tree decomposition of w. We claim that $\iota(\Xi) = \iota_0$. By the construction in Lemma 12, we know that $\iota(\Xi) = \iota_0 \circ \iota(\Xi_0)$ and the latter equals $\mathrm{id}_{Dom(\mathrm{word}(\tau))}$. Thus, $\iota(\Xi) = \iota_0$. ◀

This concludes the proof of Proposition 8: the function $\Xi \mapsto \iota(\Xi)$ is an injection by Lemma 13 and it is a surjection by Lemma 15.

▶ **Proposition 16.** *Item iii) of Theorem 5 is decidable for a finitary domain* D *given by a term* τ *over the singleton alphabet* $\{\centerdot\}$.

Proof. Assume that a term τ is given. Compute the **MSO** formula $\psi_{\mathrm{TD}(\tau)}(C_v)_{v\in\mathrm{nodes}(\tau)}$ from Proposition 10. Let φ express that there exists a unique tuple $(C_v)_{v\in\mathrm{nodes}(\tau)}$ satisfying $\psi_{\mathrm{TD}(\tau)}(C_v)_{v\in\mathrm{nodes}(\tau)}$ – we represent condensations C_v using pairs (D_v, X_v) as in Lemma 9 and use (3) to test them for equality. Apply Remark 7 to test if $D \stackrel{\mathrm{def}}{=} \mathrm{word}(\tau)$ satisfies φ. Proposition 8 implies that it is the case if and only if Item iii) of Theorem 5 holds. ◀

▶ **Corollary 17.** *If a domain* D *is finitary then the language of all words* w *such that* $Dom(w)$ *is isomorphic to* D *is regular.*

5 Uniformisations based on tree decompositions

In this section we show how to use a fixed tree decomposition Ξ of a given finitary domain D to uniformise every regular relation over D. By Proposition 8, Item iii) of Theorem 5 implies the existence of a unique such tree decomposition Ξ, which implies Item ii) of Theorem 5.

Fix a finitary domain $D = \mathrm{word}(\tau)$ for a term τ over the alphabet $\{\bullet\}$. Let $\Xi = (C_v)_{v\in\mathrm{nodes}(\tau)}$ be a fixed tree decomposition of D, represented in **MSO** by $(D_v, X_v)_{v\in\mathrm{nodes}(\tau)}$. Consider a regular synchronised relation $R \subseteq A^\circ \times B^\circ$ that is identified with a regular language $L_R \subseteq (A\times B)^\circ$. Our aim is to construct, using Ξ, a regular uniformisation of R over D.

Let $h\colon (A\times B)^\circ \to S$ recognising the language L_R with $L_R = h^{-1}(H)$. Apply the construction from [2, Lemma 29] to compute the powerset $\circ$-algebra $\mathcal{P}(S)$ with the powerset homomorphism $\mathcal{P}(h)\colon A^\circ \to \mathcal{P}(S)$, defined on the letters $a\in A$ by $\mathcal{P}(h)(a) = \big\{h\big(\binom{a}{b}\big) \mid b \in B\big\}$. The construction of $\mathcal{P}(S)$ is designed in such a way that for every word $w \in A^\circ$ we have

$$\mathcal{P}(h)(w) = \big\{h\big(\tbinom{w}{\sigma}\big) \mid \sigma \in B^{Dom(w)}\big\} \quad \text{and} \quad u \in \Pi_{A^\circ}(R) \Longleftrightarrow \mathcal{P}(h)(u)\cap H \neq \emptyset. \tag{4}$$

Notice that if $\sigma, \sigma' \in B^D$ are two words such that for every position $v \in D$ we have $h\big(\binom{w(v)}{\sigma(v)}\big) = h\big(\binom{w(v)}{\sigma'(v)}\big)$ then $(w,\sigma)\in R \Leftrightarrow (w,\sigma')\in R$. Thus, to uniformise R it is enough to choose, given a word $w\in A^\circ$, for each position $v\in D$ a type $s_v\in S$ in such a way that $s_v \in \mathcal{P}(h)(w(v))$ and $\pi\big((s_v)_{v\in D}\big)\in H$. This is summarised in the following lemma.

▶ **Lemma 18.** *If for every $s\in S$ there exists a regular uniformisation over D of the following relation denoted R_s*

$$\big\{(w,\sigma)\in \mathcal{P}(S)^\circ \times S^\circ \mid \pi(\sigma) = s \wedge Dom(w) = Dom(\sigma) \wedge \forall v \in Dom(w).\ \sigma(v)\in w(v)\big\}$$

then R also admits a regular uniformisation over D.

When the $\circ$-algebra S is *minimal* in a certain sense and one restricts in $\mathcal{P}(S)$ to the range of $\mathcal{P}(h)$ then the reciprocal of the above lemma is also true but we do not use this fact here.

From now on we work with the relations R_s's. First notice that these relations are regular themselves: the requirement that $\pi(\sigma) = s$ falls into the definition of a regular language, while the condition that $\forall v\in Dom(w).\ \sigma(v)\in w(v)$ is essentially an **MSO** sentence.

The existence of the fixed tree condensation Ξ of the domain D provides an automorphism between D and leafs(tree(τ)). Therefore, up to Ξ, we can treat w as a word over leafs(tree(τ)). Also, by (4) it is enough to construct a regular uniformisation of R_s for each $s\in S$ separately. We will now sketch an inductive construction of a uniformisation of R_s over D based on the structure of tree(τ) using the concept of *evaluation trees*. Later we will argue, that this construction can be performed in **MSO** over w based purely on Ξ.

▶ **Definition 19** ([2, Definition 7]). *Let $h\colon A^\circ \to S$ be a homomorphism into a $\circ$-monoid, τ be a term over the alphabet $\{\bullet\}$, and $D = \mathrm{word}(\tau)$. Consider a word $w\in A^D$. An* evaluation tree *of w is a labelling λ of the nodes of the condensation tree* tree(τ) *by elements of S, defined inductively by:*

- $\lambda(v) = h(w(v))$, *where v is a leaf of* tree(τ) *(indicating a subtree of the form $\bullet[]$),*
- $\lambda\big((+)[t_0,t_1]\big) = \pi\big(\lambda(t_0)\lambda(t_1)\big) = \lambda(t_0)\cdot\lambda(t_1)$,
- $\lambda\big((\Sigma_\omega)[(t_i)_{i\in\omega}]\big) = \pi\big(\lambda(t_0)\lambda(t_1)\dots\big)$,
- $\lambda\big((\Sigma_{\omega^*})[(t_i)_{i\in\omega^*}]\big) = \pi\big(\dots\lambda(t_{-3})\lambda(t_{-2})\lambda(t_{-1})\big)$.

Equivalently, one can say that $\lambda(v)$ is given by $h(w(v))$ in the leaves of tree(τ) *and if v is not a leaf and has children $(v_i)_{i\in I}$ then $\lambda(v) = \pi\big(\lambda(v_i)_{i\in I}\big)$.*

Notice that although D is finitary, $w \in A^D$ might not be finitary – this explains why we need to use the operation π instead of $(.)^\omega$ and $(.)^{\omega^\star}$. The above definition guarantees the following invariant for a node v of $\mathrm{tree}(\tau)$ and $X = \{u \in \mathrm{leafs}(\mathrm{tree}(\tau)) \mid v \preceq u\}$

$$\lambda(v) = h\big(w{\restriction}_X\big). \tag{5}$$

In particular, $\lambda(\epsilon) = h(w)$ and each word has a unique evaluation tree.

Uniformisation

Consider any element $s \in S$ and apply Theorem 4 to obtain regular uniformisations of R_s over the domains $\{0, 1\}$, ω, and $\omega^\star$. Denote these uniformisations $F_{2,s}$, $F_{\omega,s}$, and $F_{\omega^\star,s}$. We will use these uniformisations to choose types in the nodes of $\mathrm{tree}(\tau)$, producing a uniformisation F_{s_0} of R_{s_0} over D.

Recall that $D = \mathrm{leafs}(\mathrm{tree}(\tau))$ and let $w \in \mathcal{P}(S)^D$ and $\sigma \in S^D$. Let λ be the unique evaluation tree of $\left(\begin{smallmatrix} w \\ \sigma \end{smallmatrix}\right)$ in the $\circ$-semigroup $\mathcal{P}(S) \times S$ with respect to the identity homomorphism.

Let $(w, \sigma) \in F_{s_0}$ if the following conditions hold. First, for every $v \in D$ we must have $\sigma(v) \in w(v)$. Second, for $v = \epsilon$ (i.e. the root of $\mathrm{tree}(\tau)$) we must have $\lambda(v) = (T, s)$ with $s = s_0$. Finally, consider any node $v \in \mathrm{nodes}(\mathrm{tree}(\tau))$ that is not a leaf, let $\lambda(v) = (T, s)$, and assume that $(v_i)_{i\in I}$ are the children of v in $\mathrm{tree}(\tau)$. Let $\left(\begin{smallmatrix} w' \\ \sigma' \end{smallmatrix}\right) = \big(\lambda(v_i)\big)_{i\in I}$ be the word over $\mathcal{P}(S) \times S$ obtained by taking the λ-values of the children of v. Then we must have that if v is labelled by $(+)$ (resp. $(\times\omega)$ or $(\times\omega^\star)$), then (w', σ') belongs to $F_{2,s}$ (resp. $F_{\omega,s}$ or $F_{\omega^\star,s}$).

▶ **Lemma 20.** *For every $s_0 \in S$ the relation F_{s_0} is a uniformisation over D of R_{s_0}.*

A proof of this lemma is based on induction over $\mathrm{tree}(\tau)$ and repetitive usage of the fact that the relations $F_{2,s}$, $F_{\omega,s}$, and $F_{\omega^\star,s}$ are uniformised.

▶ **Lemma 21.** *For each $s \in S$ the relation F_s is regular with parameter Ξ: there exists an **MSO**-formula $\psi_{F_s}\big((D_v, X_v)_{v\in\mathrm{nodes}(\tau)}\big)$ over the alphabet $\mathcal{P}(S)\times S$ which holds over a given word $\left(\begin{smallmatrix} w \\ \sigma \end{smallmatrix}\right)$ with parameters $(D_v, X_v)_{v\in\mathrm{nodes}(\tau)}$ if and only if $(D_v, X_v)_{v\in\mathrm{nodes}(\tau)}$ represents a tree decomposition Ξ with shape τ of w and $(w, \sigma) \in F_s$ where the relation F_s is defined as above based on Ξ.*

The construction is based on the fact that the tree decomposition Ξ provides a way to **MSO**-encode the structure of $\mathrm{tree}(\tau)$ over the given word w. This makes the definition of F_s definable in **MSO** over (w, σ).

This concludes the proof of the implication *iii)* ⇒ *ii)* of Theorem 5: if there is a unique automorphism of w then there is a unique tree decomposition Ξ_0 of w that can be fixed in **MSO** using the formula $\psi_{\mathrm{TD}(\tau)}$ from Proposition 10.

6 Conclusions

The main result of this work shows that in the case of countable domains, the only obstacle for regular uniformisations are non-trivial automorphisms. This provides a very clean picture: given a domain D, either all regular relations over D have regular uniformisations, or already the simple relation of choice over D has no regular uniformisation because the domain D admits *shifts* (non-trivial automorphisms).

The techniques involved in the proof of this result are based mainly on the tools developed in [2] to study the algebraic structure of regular languages of countable words. However, one needs to carefully merge tools coming from logic and algebra to actually construct regular uniformisations under the assumption of lack of shifts. This is achieved by showing that in the considered setup, one can encode evaluation trees from [2] within **MSO**. That approach differs from the one taken in [2] when moving from algebra to logic, because there the shape of the domain of the word is unknown.

A possible next step on our way of understanding uniformisability is to generalise the present result with that of [4]: given a particular relation R over countable words, decide if R admits a regular uniformisation. To achieve that, one should understand how to merge the techniques of [4] that analyse the case of words over $\mathbb{Z}$; with the above results clarifying the situation under the assumption of "no interval of the form $I \times \mathbb{Z}$".

References

1. Arnaud Carayol and Christof Löding. MSO on the infinite binary tree: Choice and order. In *CSL*, pages 161–176, 2007.
2. Olivier Carton, Thomas Colcombet, and Gabriele Puppis. An algebraic approach to **MSO**-definability on countable linear orderings. *J. Symb. Log.*, 83(3):1147–1189, 2018.
3. Heinz-Dieter Ebbinghaus, Jörg Flum, and Wolfgang Thomas. *Mathematical Logic*. Undergraduate Texts in Mathematics. Springer New York, 1996.
4. Grzegorz Fabiański, Michał Skrzypczak, and Szymon Toruńczyk. Uniformisations of regular relations over bi-infinite words. In *LICS*, pages 384–396, 2020.
5. Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.
6. Yuri Gurevich and Saharon Shelah. Rabin's uniformization problem. *Journal of Symbolic Logic*, 48(4):1105–1119, 1983.
7. Shmuel Lifsches and Saharon Shelah. Uniformization and Skolem functions in the class of trees. *Journal of Symbolic Logic*, 63(1):103–127, 1998.
8. Anil Nerode. Linear automaton transformations. *Proceedings of the American Mathematical Society*, 9(4):541–544, 1958.
9. Alexander Rabinovich. On decidability of monadic logic of order over the naturals extended by monadic predicates. *Information and Computation*, 205(6):870–889, 2007.
10. Dirk Siefkes. The recursive sets in certain monadic second order fragments of arithmetic. *Arch. Math. Logik*, 17(1–2):71–80, 1975.