Regular Languages of Words Over Countable Linear Orderings

Olivier Carton[1], Thomas Colcombet[2], and Gabriele Puppis[3]

[1] University Paris Diderot, LIAFA
olivier.carton@liafa.jussieu.fr
[2] CNRS/LIAFA
thomas.colcombet@liafa.jussieu.fr
[3] Oxford University Computing Laboratory
gabriele.puppis@comlab.ox.ac.uk

Abstract. We develop an algebraic model suitable for recognizing languages of words indexed by countable linear orderings. We prove that this notion of recognizability is effectively equivalent to definability in monadic second-order (MSO) logic. This reproves in particular the decidability of MSO logic over the rationals with order. Our proof also implies the first known collapse result for MSO logic over countable linear orderings.

1 Introduction

This paper continues a long line of research aiming at understanding the notions of regularity for languages of infinite objects, i.e., of infinite words and trees. This research results both in decision procedures for the *monadic second-order* (*MSO*) *logic* and in a fine comprehension of the mechanisms involved in different models of recognition. More specifically, the paper answers the following interrogations:

> What is a good notion of regularity for languages of words indexed by countable linear orderings? Is it equivalent to definability in MSO? What are the correct tools for studying this notion?

Several results, in particular in terms of decidability, partially answered the above questions (see related work below). Our study however gives a deeper insight in the phenomena, for instance, answering (positively) the following previously open question:

> Does there exist a collapse result for MSO logic over countable linear orders, as Büchi's result shows a collapse of MSO logic to its existential fragment for words indexed by ω?

The central objects in this paper are *words* indexed by countable linear orderings, i.e., total orders over countable sets together with applications mapping elements to letters in some finite alphabet. Languages are just sets of countable words and MSO logic gives a formalism for describing such languages in terms of formulas involving quantification over elements and sets of elements (a formula naturally defines the language of all words that makes the formula true).

Related work. Büchi initiated the study of MSO logic using the tools of language theory. In particular, he established that every language of ω-words (i.e., the particular case of words indexed by the ordinal ω) definable in MSO logic is effectively accepted by a suitable form of automata [5]. A major advance has been obtained by Rabin, who extended this result to infinite trees [8]. One consequence of Rabin's result is that MSO logic is decidable over the class of all countable linear orderings. Indeed, every linear ordering can be seen as some set of nodes of the infinite binary tree, the linear order corresponding to the infix ordering on nodes. Another proof of the decidability of the MSO theory of countable orders has been proposed by Shelah using the composition method [12]. This method is an automaton-free approach to logic based on syntactic operations on formulas and inspired from Feferman and Vaught [7]. The same paper of Shelah is also of major importance for another result it contains: the undecidability of the MSO theory of the real line (the reals with order).

However, for ω-words as for infinite trees, the theory is much richer than simply the decidability of MSO logic. In particular, MSO logic is known to be equivalent to several formalisms, such as automata and, in the ω-word case, regular expressions and some forms of algebras, which admit minimization and give a very deep insight in the structure of languages. The decidability proof for MSO logic over countable words does not provide such an understanding.

A branch of research has been pursued to raise the equivalence between logic, automata, and algebra to infinite words beyond ω-words. In [4], Büchi introduced ω_1-automata on transfinite words to prove the decidability of MSO logic for ordinals less than ω_1. Besides the usual transitions, ω_1-automata are equipped with limit transitions of the form $P \to q$, with P set of states, which are used in a Muller-like way to process words indexed over ordinals. Büchi proved that his automata have the same expressive power as MSO logic for ordinals less than ω_1. The key ingredient is the closure under complementation of ω_1-automata.

In [2], ω_1-automata have been extended to $\diamond$-automata by introducing limit transitions of the form $q \to P$ to process words over linear orderings. In [10], $\diamond$-automata are proven to be closed under complementation with respect to countable and scattered orderings. This last result implies that $\diamond$-automata have the same expressive power as MSO logic over countable and scattered orderings [1]. However, it was already noticed in [1] that $\diamond$-automata are strictly weaker than MSO logic over countable (possibly non-scattered) linear orderings: indeed, the closure under complementation fails as there is an automaton that accepts all words with non-scattered domains, whereas there is none for scattered words.

In this paper, we unify those branches of research. We provide an algebraic framework and a notion of recognizability which happens to be equivalent to the definability in MSO logic. Our approach both extends the decidability approach of Rabin and Shelah, and provides new results concerning the expressive power of MSO logic over countable linear orders. In preliminary versions of this work, we devised an equivalent automaton model. This notion is less natural and it is not presented in this short paper.

Structure of the paper. After some definitions in Section 2, we present the algebraic models of $\circ$-algebras in Section 3 and describe the corresponding tools and results. In Section 4 we translate MSO formulas to $\circ$-algebras and in Section 5 we establish the converse.

2 Preliminaries

Linear orderings. A *linear ordering* $\alpha = (X, <)$ is a non-empty set X equipped with a total order $<$. Two linear orderings have same *order type* if there is an order-preserving bijection between their domains. We denote by ω, ω^*, ζ, η the order types of $(\mathbb{N}, <)$, $(-\mathbb{N}, <)$, $(\mathbb{Z}, <)$, $(\mathbb{Q}, <)$, respectively. Unless strictly necessary, we do not distinguish between a linear ordering and its order type.

A *sub-ordering* of α is a subset I of α equipped with the same ordering relation (we denote it by $\alpha|_I$). Given two subsets I, J of α, we write $I < J$ iff $x < y$ for all $x \in I$ and all $y \in J$. A subset I of α is said to be *convex* if for all $x, y \in I$ and all $z \in \alpha$, $x < z < y$ implies $z \in I$.

A linear ordering α is *dense* if for all $x < y \in \alpha$, there is $z \in \alpha$ such that $x < z < y$. It is *scattered* if none of its sub-orderings is both dense and non-singleton.

The *sum* $\alpha_1 + \alpha_2$ of two linear orderings $\alpha_1 = (X_1, <_1)$ and $\alpha_2 = (X_2, <_2)$ (up to renaming, assume that X_1 and X_2 are disjoint) is the linear ordering $(X_1 \uplus X_2, <)$, where $<$ coincides with $<_1$ on X_1, with $<_2$ on X_2, and, furthermore, it satisfies $X_1 < X_2$. More generally, given a linear ordering $\alpha = (X, <)$ and, for each $i \in X$, a linear ordering $\beta_i = (Y_i, <_i)$ (assume that the sets Y_i are pairwise disjoint), we denote by $\sum_{i \in \alpha} \beta_i$ the linear ordering $(Y, <')$, where $Y = \bigcup_{i \in X} Y_i$ and, for every $i, j \in X$, every $x \in Y_i$, and every $y \in Y_j$, $x <' y$ iff either $i = j$ and $x <_i y$ hold or $i < j$ holds.

Additional material on linear orderings can be found in [11].

Condensations. A standard way to prove properties of linear orderings is to decompose them into basic components (e.g., finite sequences, ω-sequences, ω^*-sequences, and η-orderings). This can be done by exploiting the notion of condensation. Precisely, a *condensation* of a linear ordering α is an equivalence relation $\sim$ over it such that for all $x < y < z$ in α, $x \sim z$ implies $x \sim y \sim z$ (this is equivalent to enforcing the condition that every equivalence class of $\sim$ is a convex subset). The ordering relation of α induces a corresponding total order on the *quotient* $\alpha/\!\sim$, which is called *condensed order*. Conversely, any partition C of α into convex subsets induces a condensation $\sim_C$ such that $x \sim y$ iff x and y belong to the same convex subset $I \in C$.

Words and languages. We use a generalized notion of word, which coincides with the notion of labeled linear ordering. Given a linear ordering α and a finite alphabet A, a *word over A of domain α* is a mapping of the form $w : \alpha \to A$. Hereafter, we shall consider words up to isomorphism of the domain, unless a specific presentation of the domain is required. Moreover, we are only interested in words of countable domains. The set of words over alphabet A

is denoted A°. Given a word w of domain α and a non-empty subset I of α, we denote by $w|_I$ the *sub-word* resulting from the restriction of the domain of w to I. Furthermore, if I is convex, then $w|_I$ is said to be a *factor* of w.

Certain words will play a crucial role in the sequel. In particular, a word $w : \alpha \to A$ is said to be a *perfect shuffle of* A if (i) the domain α is isomorphic to $\mathbb{Q}$ and (ii) for every symbol $a \in A$, the set $w^{-1}(a) = \{x \in \alpha \mid w(x) = a\}$ is dense in α. Recall that $\mathbb{Q}$ is the unique, up to isomorphism, countable non-singleton dense linear ordering with no end-points. Likewise, for every finite set A, there is a unique, up to isomorphism, perfect shuffle of A.

Given two words $u : \alpha \to A$ and $v : \beta \to A$, we denote by uv the word of domain $\alpha + \beta$ where each position $x \in \alpha$ (resp., $x \in \beta$) is labeled either by $u(x)$ (resp., $v(x)$). The concatenation of words is easily generalized to the infinite concatenation $\prod_{i\in\alpha} w_i$, where α is a linear ordering and each w_i has domain β_i, the result being a word of domain $\sum_{i\in\alpha} \beta_i$. We also define the *shuffle* of a tuple of words $w_1, ..., w_k$ as the word $\{w_1, ..., w_k\}^\eta = \prod_{i\in\mathbb{Q}} w_{f(i)}$, where f is the unique perfect shuffle of $\{1, ..., k\}$ of domain $\mathbb{Q}$.

A *language* is a set of words over a certain alphabet. For some technical reasons, it is convenient to avoid the presence of the empty word ε in a language. Thus, unless otherwise specified, we use the term word to mean a labeled, countable, *non-empty* linear ordering. The operations of juxtaposition, ω-iteration, ω^*-iteration, shuffle, etc. are extended to languages in the obvious way.

3 Semigroups and algebras for countable linear orderings

This section is devoted to the presentation of algebraic objects suitable for defining a notion of recognizable $\circ$-languages. As it is already the case for ω-words, our definitions come in two flavors, $\circ$-semigroups (corresponding to ω-semigroups) and $\circ$-algebras (corresponding to Wilke-algebras). We prove the equivalence of the two notions when the underlying set is finite.

Countable products. The objective is to have a notion of products indexed by countable linear orderings, and possessing several desirable properties (in particular, generalized associativity and existence of finite presentations).

Definition 1. *A* (generalized) product *over a set S is an application π from S° to S such that, for every $a \in S$, $\pi(a) = a$ and, for every word u and every condensation $\sim$ of its domain,*

$$\pi(u) \;=\; \pi\Big(\prod_{I\in\alpha/\sim} \pi(u|_I)\Big) \qquad \text{(generalized associativity)}$$

The pair $\langle S, \pi\rangle$ is called a $\circ$-semigroup*.*

As an example, the operation $\prod$ is a generalized product over A°. Hence, $\langle A^\circ, \prod\rangle$ is a $\circ$-semigroup, called the *free $\circ$-semigroup generated by A*.

A *morphism* from a $\circ$-semigroup (S, π) to another $\circ$-semigroup (S', π') is a mapping $\varphi : S \to S'$ such that, for every word $w : \alpha \to S$, $\varphi(\pi(w)) = \pi'(\tilde{\varphi}(w))$, where $\tilde{\varphi}$ is the component-wise extension of φ to words. A $\circ$-language

$L \subseteq A^\circ$ is said *recognizable by ∘-semigroup* if there exists a morphism φ from $\langle A^\circ, \prod \rangle$ to some finite semigroup $\langle S, \pi \rangle$ (here finite means that S is finite) such that $L = \varphi^{-1}(F)$ for some $F \subseteq S$ (equivalently, $\varphi^{-1}(\varphi(L)) = L$).

Recognizability by ∘-semigroup has the expressive power we aim at, however, the product π requires to be represented, a priori, by an infinite table. This is not usable as it stands for decision procedures. That is why, given a finite ∘-semigroup $\langle S, \pi \rangle$, we define the following (finitely presentable) algebraic operators:

- $\cdot : S^2 \rightarrow S$, which maps a pair of elements $a, b \in S$ to the element $\pi(ab)$,
- $\tau : S \rightarrow S$, which maps an element $a \in S$ to the element $\pi(a^\omega)$,
- $\tau^* : S \rightarrow S$, which maps an element $a \in S$ to the element $\pi(a^{\omega^*})$,
- $\kappa : \mathscr{P}(S) \rightarrow S$, which maps a non-empty set $\{a_1, ..., a_k\}$ to $\pi\big(\{a_1, ..., a_k\}^\eta\big)$.

One says that $\cdot, \tau, \tau^*$ and κ are *induced by* π. From now on, we shall use the operator $\cdot$ with infix notation (e.g., $a \cdot b$) and the operators τ, τ^*, and κ with superscript notation (e.g., a^τ, $\{a_1, ..., a_k\}^\kappa$). The resulting algebraic structure $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$ has the property of being a ∘-algebra, defined as follows:

Definition 2. *A structure $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$ in which $\cdot : S^2 \rightarrow S$, $\tau, \tau^* : S \rightarrow S$ and $\kappa : \mathcal{P}(S) \rightarrow S$, is called a* ∘-algebra *if:*

(A1) $(S, \cdot)$ *is a semigroup, namely, for every $a, b, c \in S$, $a \cdot (b \cdot c) = (a \cdot b) \cdot c$,*

(A2) τ *is* compatible to the right*, namely, for every $a, b \in S$ and every $n > 0$, $(a \cdot b)^\tau = a \cdot (b \cdot a)^\tau$ and $(a^n)^\tau = a^\tau$,*

(A3) τ^* *is* compatible to the left*, namely, for every $a, b \in S$ and every $n > 0$, $(b \cdot a)^{\tau^*} = (a \cdot b)^{\tau^*} \cdot a$ and $(a^n)^{\tau^*} = a^{\tau^*}$,*

(A4) κ *is* compatible with shuffles*, namely, for every non-empty subset P of S, every element c in P, every subset Q of P, and every non-empty subset R of $\{P^\kappa, a \cdot P^\kappa, P^\kappa \cdot b, a \cdot P^\kappa \cdot b \ : \ a, b \in P\}$, we have $P^\kappa = P^\kappa \cdot P^\kappa = P^\kappa \cdot c \cdot P^\kappa = (P^\kappa)^\tau = (P^\kappa \cdot c)^\tau = (P^\kappa)^{\tau^*} = (c \cdot P^\kappa)^{\tau^*} = (Q \cup R)^\kappa$.*

The typical ∘-algebra is:

Lemma 1. *For all A, $\langle A^\circ, \cdot, \omega, \omega^*, \eta \rangle$ is a ∘-algebra*[4]*.*

Proof. By a systematic analysis of Axioms A1-A4.

Furthermore, as we mentioned above, every ∘-semigroup induces a ∘-algebra.

Lemma 2. *For all finite ∘-semigroup $\langle S, \pi \rangle$, $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$ is a finite ∘-algebra, where the operators $\cdot, \tau, \tau^*$, and κ are those induced by π.*

Proof. The result is simply inherited from Lemma 1 by morphism. Let $\langle S, \pi \rangle$ be a ∘-semigroup, which defines $\cdot, \tau, \tau^*, \kappa$. The structure $\langle S^\circ, \prod \rangle$ is also a ∘-semigroup, which defines concatenation, ω, ω^* and η. Furthermore, the product π can also

[4] This ∘-algebra is not the free ∘-algebra generated from A. The free ∘-algebra generated from a finite set is by definition countable, while A° has the cardinality of the continuum. This situation is similar to the one of Wilke-algebras.

be seen as a surjective morphism from $\langle S^\circ, \prod\rangle$ to $\langle S, \pi\rangle$ (just a morphism of algebras, not of $\circ$-algebras). By definition of $\cdot, \tau, \tau^*, \kappa$, this morphism maps concatenation to $\cdot$, ω to τ, ω^* to τ^*, and η to κ. It follows that any equality involving concatenation, ω, ω^* and η is also satisfied by $\langle S, \cdot, \tau, \tau^*, \kappa\rangle$ with concatenation replaced by $\cdot$, ω by τ, ω^* by τ^*, and η by κ. In particular all axioms of $\circ$-algebras which hold by Lemma 1 are directly transfered to $\langle S, \cdot, \tau, \tau^*, \kappa\rangle$. $\square$

Extension of $\circ$-algebras to countable products. Here, we aim at proving a converse to Lemma 2, namely, that every *finite* $\circ$-algebra $\langle S, \cdot, \tau, \tau^*, \kappa\rangle\rangle$ can be uniquely extended into a unique $\circ$-semigroup $\langle S, \pi\rangle$ (Theorem 1). We assume that all words are over the alphabet S. Moreover, we denote by ε the empty word, with the convention that $\varepsilon \notin S$ and $a \cdot \varepsilon = \varepsilon \cdot a = a$ for all $a \in S \cup \{\varepsilon\}$ (note that we distinguish the empty word ε from the identity element of S, if there is any). The objective of the construction is to attach to each word u over S a 'value' in S. Furthermore, this value needs to be unique.

The central objects in this proof are *evaluation trees*, i.e., infinite trees describing how a word in S° can be evaluated into an element of S. We begin with condensation trees which are convenient representations for nested condensations. The nodes of a condensation tree are convex subsets of the linear ordering and the descendant relation is the inclusion. The set of children of each node defines a condensation. Furthermore, in order to provide an induction parameter, we require that the branches of a condensation tree are finite (but their length may not be uniformly bounded).

Definition 3. *A* condensation tree *over a linear ordering α is a set T of non-empty convex subsets of α such that:*

- *$\alpha \in T$,*
- *for all I, J in T, either $I \subseteq J$ or $J \subseteq I$ or $I \cap J = \emptyset$,*
- *for all $I \in T$, the union of all $J \in T$ such that $J \subsetneq I$ is either I or $\emptyset$,*
- *every subset of T totally ordered by inclusion is finite.*

Elements in T are called *nodes*. The node α is called the *root* of the tree. Nodes minimal for $\subseteq$ are called *leaves*. Given $I, J \in T$ such that $I \subsetneq J$ and there exist no $K \in T$ such that $I \subsetneq K \subsetneq J$, then I is called a *child* of J and J the *parent* of I. According to the the definition, if I is an *internal node*, i.e., is not a leaf, then it has a set of children $\texttt{children}_T(I)$ which form a partition of I. This partition consisting of convexes, it corresponds naturally to a condensation of $\alpha|_I$. When the tree T is clear from the context, we will denote by $\texttt{children}(I)$ the set of all children of I in T and, by extension, the corresponding condensation and the corresponding condensed linear ordering.

Since the branches of a condensation tree are finite, an ordinal rank can be associated with such a tree. This is the smallest ordinal β that enables a labeling of the nodes by ordinals less than or equal to β such that the label of each node is strictly greater than the labels of its children. This rank allows us to make proofs by induction (see also [**?**] for similar definitions).

We now introduce evaluation trees. Intuitively, these are condensation trees whose nodes have an associated value in S and such that the value of a node can be 'easily' computed from the value of its children. What we mean by being 'easy to compute' is precisely what has a value for the following mapping π_0:

Definition 4. *Let π_0 be the partial mapping from S° to S such that:*

- $\pi_0(s_1 ... s_n) = s_1 \cdot ... \cdot s_n$ *for all* $n \geq 1$ *and all* $s_1, ..., s_n \in S$,
- $\pi_0(se^\omega) = s \cdot e^\tau$ *for all* $s, e \in S$ *with* e *idempotent,*
- $\pi_0(e^{\omega^*} s) = e^{\tau^*} \cdot s$ *for all* $s, e \in S$ *with* e *idempotent,*
- $\pi_0(sP^\eta t) = s \cdot P^\kappa \cdot t$ *for all* $s, t \in S \cup \{\varepsilon\}$ *and all non-empty sets* $P \subseteq S$,
- *in any other case* π_0 *is undefined.*

Definition 5. *An* evaluation tree *over a linear ordering* α *is a pair* $\mathcal{T} = \langle T, \gamma \rangle$ *in which* T *is a condensation tree over* α, *and* γ *is a mapping from* T *to* S *such that for every internal node* $I \in T$, $\pi_0(\gamma(\texttt{children}(I)) = \gamma(I)$ *(in particular it is defined), in which* $\gamma(\texttt{children}(I))$ *denotes the word of domain* $\texttt{children}(I)$ *where each position* $J \in \texttt{children}(I)$ *is labeled by* $\gamma(J)$*). The* value *of* $\langle T, \gamma \rangle$ *is* $\gamma(\alpha)$, *i.e., the value of the root.*
An evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ *over a word* u *is an evaluation tree over the domain of* u *such that the leaves of* T *are singletons, and* $\gamma(\{x\}) = u(x)$ *for all* x *in the domain of* u.

The next propositions are central in the study of evaluation trees.

Proposition 1. *For every word* u, *there exists an evaluation tree over* u.

The proof here resembles the construction used by Shelah in his proof of decidability of the monadic second-order theory of orders from [12]. In particular, it uses the theorem of Ramsey, as well as a lemma stating that every non-trivial word indexed by a dense linear ordering has a perfect shuffle as a factor. We remark that the above proposition does not use any of the Axioms A1-A4.

Proposition 2. *Two evaluation trees over the same word have same value.*

The proof of this result is quite involved and it heavily relies on the use of Axioms A1-A4 (each axiom can be seen as an instance of Proposition 2 in some special cases of computation trees of height 2). The proof makes also use of Proposition 1. Note that, as opposed to Proposition 1, Proposition 2 has no counterpart in [12].

Using the above results, the proof of the following result is relatively easy:

Theorem 1. *For every finite* $\circ$*-algebra* $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$, *there exists a product* π *which defines* $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$.

Proof. Given a word w of domain α, one defines $\pi(w)$ to be the value of some evaluation tree over w (the evaluation tree exists by Proposition 1 and the value $\pi(w)$ is unique by Proposition 2).

We prove that π is associative. Let $\sim$ be a condensation of the domain α. For all $I \in \alpha/\sim$, let $\mathcal{T}_I$ be some evaluation tree over $w|_I$. Let also $\mathcal{T}'$ be some evaluation tree over the word $w' = \prod_{I\in\alpha/\sim} \pi(w|_I)$. One constructs an evaluation tree $\mathcal{T}$ over w by substituting[5] each leaf of $\mathcal{T}'$ corresponding to some class $I \in \alpha/\sim$ with the subtree $\mathcal{T}_I$. This is possible (i.e., respects the definition of evaluation tree) since the value of each evaluation tree $\mathcal{T}_I$ is $\pi(w|_I)$, which coincides with the value $w'(I)$ at the leaf I of $\mathcal{T}'$. By Proposition 2, the resulting evaluation tree $\mathcal{T}$ has the same value as $\mathcal{T}'$ and this witnesses that $\pi(w) = \pi\left(\prod_{I\in\alpha/\sim} \pi(w|_I)\right)$.

What remains to be done is to prove that indeed the above choice of π defines $\cdot, \tau, \tau^*, \kappa$. This requires a case by case analysis. $\square$

Let us conclude with a decidability result.

Theorem 2. *Emptiness of $\circ$-languages recognizable by $\circ$-algebras is decidable.*

Proof (principle of the algorithm). The algorithm takes the image of letters in the $\circ$-algebra, and saturates it by $\cdot, \tau, \tau^*$, and κ. It answers yes if the resulting set intersects the accepting subset of the algebra, and no otherwise. $\square$

4 From monadic second-order logic to $\circ$-algebras

Let us recall that monadic second-order (MSO) logic is the extension of first-order logic with set quantifiers. We assume the reader to have some familiarity with this logic as well as with the Büchi approach for translating MSO formulas into automata. A good survey can be found in [13]. We refer to the $\forall$-*fragment* as the set of formulas that start with a block of universal set quantifiers, followed by a first-order formula. The $\exists\forall$-*fragment* consists of formulas starting with a block of existential set quantifiers followed by a formula of the $\forall$-fragment.

Here, we mimic Büchi's technique and show a relatively direct consequence of the above results, namely that MSO formulas can be translated to $\circ$-algebras:

Proposition 3. *The MSO definable languages are effectively $\circ$-recognizable.*

Let us remark that we could have equally well used the composition method of Shelah for establishing Proposition 3. Indeed, given an MSO definable language, a $\circ$-algebra recognizing it can be directly extracted from [12].

Our chosen proof for Proposition 3 follows Büchi's approach, namely, we establish sufficiently many closure properties of $\circ$-recognizable language. Then, each construction of the logic can be translated into an operation on languages. To disjunction corresponds union, to conjunction corresponds intersection, to negation corresponds complement, etc. We assume the reader to be familiar with this approach (in particular the coding of the valuations of free variables).

The closure under intersection, union, and complement are, as usual, easy to obtain. The languages corresponding to atomic predicates are also very easily

[5] The proof is not correct as this. An exact proof requires also to raise $\mathcal{T}'$ from the linear ordering $\alpha/\sim$ to α. This is done by replacing each node J in $\mathcal{T}'$ by $\bigcup J$.

shown to be $\circ$-recognizable. What remains to be proved is the closure under projection. Given a language of $\circ$-words L over some alphabet A, and a mapping h from A to another alphabet B, the *projection of L by h* is simply $h(L)$ (h being extended component-wise to $\circ$-words, and $\circ$-languages). It is classical that this projection operation is what is necessary for obtaining the closure under existential quantification at the logical level. Hence, we just need to prove:

Lemma 3. *The $\circ$-recognizable languages are effectively closed under projections.*

Proof (sketch). We first describe the construction for a $\circ$-semigroup $\langle S,\pi\rangle$. The projection is obtained, as it is usual, by a powerset construction, i.e., we aim at providing a $\circ$-product over $\mathcal{P}(S)$. Given two words u and U over S and $\mathcal{P}(S)$ respectively, we write $u \in U$ when $\mathcal{D}om(u) = \mathcal{D}om(U)$ and $u(x) \in U(x)$ for all $x \in \mathcal{D}om(U)$. We define the mapping $\tilde{\pi}$ from $(\mathcal{P}(S))^\circ$ to $\mathcal{P}(S)$ by

$$\tilde{\pi}(U) \;=^{\text{def}}\; \{\pi(u) \;:\; u \in U\} \qquad \text{for all } U \in (\mathcal{P}(S))^\circ .$$

Let us show that $\tilde{\pi}$ is associative. Consider a word U over $\mathcal{P}(S)$ and a condensation $\sim$ of its domain. Then,

$$\begin{aligned}\tilde{\pi}(U) \;&=\; \Big\{\pi(u) \;:\; u \in U\Big\} \;=\; \Big\{\pi\Big(\textstyle\prod_{I\in\alpha/\sim}\pi(u|_I)\Big) \;:\; u \in U\Big\}\\ &=\; \Big\{\pi\Big(\textstyle\prod_{I\in\alpha/\sim} a_I\Big) \;:\; a_I \in \tilde{\pi}(U|_I) \text{ for all } I \in \alpha/\sim\Big\} \;=\; \tilde{\pi}\Big(\textstyle\prod_{I\in\alpha/\sim}\tilde{\pi}(U|_I)\Big)\,,\end{aligned}$$

where the second equality is derived from the associativity of π. Hence $(\mathcal{P}(S),\tilde{\pi})$ is a $\circ$-semigroup. It is just a matter a writing to show that $\langle\mathcal{P}(S),\tilde{\pi}\rangle$ recognizes any projection of a language recognized by $\langle S,\pi\rangle$.

Of course, thanks to Lemma 2 and Theorem 1, this construction can be performed at the level of $\circ$-algebras. The problem is that this may, a priori, be non-effective. However, using a more careful analysis, it is possible to prove the effectiveness of the construction. $\square$

5 From $\circ$-algebras to monadic second-order logic

We have seen in the previous section that every MSO formula defines a $\circ$-recognizable language. In this section, we sketch the proof of the converse.

Theorem 3. *Every $\circ$-recognizable $\circ$-language is effectively MSO definable. Furthermore, every such language is definable in the $\exists\forall$-fragment of MSO logic.*

We fix for the remaining of the section a morphism h from $\langle A^\circ,\prod\rangle$ to a $\circ$-semigroup $\langle S,\pi\rangle$, with S finite. Let F be some subset of S. Let also $\cdot,\tau,\tau^*,\kappa$ be defined from π. Our goal is to show that $L = h^{-1}(F)$ is MSO definable. It is sufficient for this to show that for every $s \in S$, the language

$$\pi^{-1}(s) \;=\; \{w \in S^\circ \;:\; \pi(w) = s\}\;,$$

is defined by some MSO formula φ_s. This establishes that $L = \bigcup_{s\in F} h^{-1}(s)$ is defined by the disjunction $\bigvee_{s\in F}\varphi'_s$, where φ'_s is obtained from φ_s by replacing every occurrence of an atom $t(x)$, with $t \in S$, by $\bigvee_{a\in h^{-1}(t)\cap A} a(x)$.

A good approach for defining $\pi^{-1}(s)$ is to use a formula that, given $w \in S^\circ$, would guess some object 'witnessing' $\pi(w) = s$. The only objects that we have seen so far and that are able to 'witness' $\pi(w) = s$ are evaluation trees. Unfortunately, there is no way an MSO formula can guess an evaluation tree, since their height cannot be uniformly bounded. That is why we use another kind of objects for witnessing $\pi(w) = a$: the Ramsey splits, introduced just below.

Ramsey splits. Ramsey splits are not directly applied to words, but to additive labellings. An *additive labeling* σ from a linear order α to a semigroup $\langle S, \cdot \rangle$ (in particular, this will be a $\circ$-semigroup in our case) is a function that maps any pair of elements $x < y$ from α to an element $\sigma(x, y) \in S$ in such a way that $\sigma(x, y) \cdot \sigma(y, z) = \sigma(x, z)$ for all $x < y < z$ in α.

Given two positions x, y in a word w, denote by $[x, y)$ the interval $\{z \ : \ x \leq z < y\}$. Given a word w and two positions $x < y$ in it, we define $\sigma_w(x, y) \in S$ to be $\pi(w|_{[x,y)})$. We just mention σ whenever w is clear from the context. Quite naturally, σ_w is additive since for all $x < y < z$, we have $\sigma(x, y) \cdot \sigma(y, z) = \pi\big(w|_{[x,y)}\big) \cdot \pi\big(w|_{[y,z)}\big) = \pi\big(w|_{[x,y)} w|_{[y,z)}\big) = \pi\big(w|_{[x,z)}\big) = \sigma(x, z)$.

Definition 6. *A* split *of a linear ordering α of height n is a function $g : \alpha \to [1, n]$. Two elements $x < y$ in α are called* (k-)neighbors *iff $g(x) = g(y) = k$ and $g(z) \leq k$ for all $z \in [x, y]$ (note that neighborhood is an equivalence relation).*
The split g is called Ramsey *for some additive labeling σ iff for all equivalence classes $X \subseteq \alpha$ for the neighborhood relation, there is an idempotent $e \in S$ such that $\sigma(x, y) = e$ for all $x < y$ in X.*

Theorem 4 (Colcombet [6]). *For every finite semigroup $\langle S, \cdot \rangle$, every linear ordering α, and every additive labeling σ from α to $\langle S, \cdot \rangle$, there is a split of α which is Ramsey for σ and which has height at most $2|S|$.*

From $\circ$-recognizable to MSO definable. The principle is to construct a formula which, given a word w, guesses a split of height at most $2|S|$, and use it for representing the application which to every convex set I associates $\pi(w|_I)$. For the explanations, we assume that some word w is fixed, that its domain is α, and that σ is the additive labeling over α derived from w. We remark, however, that all constructions are uniform and do not depend on w.

We aim at constructing a formula $\texttt{evaluate}_s$, for each $s \in S$, which holds over a word w iff $\pi(w) = s$. The starting point is to guess the two following pieces of information:

- a split g of α of height at most $2|S|$,
- an application e mapping each position $x \in \alpha$ to an idempotent of S.

The intention is that a good choice of g, e by the formula is when the split g is Ramsey for σ and the application e maps each x to the idempotent $e(x)$ that arises when the neighborhood class of x is considered in the definition of Ramseyness. In such a case, we say that (g, e) is Ramsey. This is an advance toward computing the value of a word, since Ramsey splits can be used as 'accelerating structures' in the sense that every computation of some $\pi(w|_I)$ for a convex subset I becomes significantly easier when a Ramsey split is known. In particular, computing $\pi(w|_I)$, for any convex subset I, becomes first-order definable!

Observe that neither g nor e can be represented by a single monadic variable. However, since both g and e are mappings from α to sets of bounded size ($2|S|$ for g, and $|S|$ for e), one can guess them using a fixed number of monadic variables. This kind of coding is standard, and from now on we shall use explicitly the mappings g and e in MSO formulas, rather than their codings.

Lemma 4. *For all $s \in S$, there is a first-order formula* $\mathtt{value}_s(g, e, X)$*, such that for every convex subset I:*

- *if (g, e) is Ramsey, then* $\mathtt{value}_s(g, e, I)$ *holds iff $\pi(w|_I) = s$,*
- *if both* $\mathtt{value}_s(g, e, I)$ *and* $\mathtt{value}_t(g, e, I)$ *hold, then $s = t$.*

One sees those formulas as defining a partial function `value` mapping g, e, I to some element $s \in S$ (the second item enforces that there is no ambiguity about the value, namely, that this is a function and not a relation). From now we simply use the notation $\mathtt{value}(g, e, I)$ as if it was a function.

One needs now to enforce that $\mathtt{value}(g, e, I)$ coincides with $\pi(w|_I)$, even without assuming that (g, e) is Ramsey. For this, one uses condensations. A priori, a condensation is not representable by monadic variables, since it is a binary relation. However, any set $X \subseteq \alpha$ naturally defines the relation $\approx_X$ such that $x \approx_X y$ iff either $[x, y] \subseteq X$, or $[x, y] \cap X = \emptyset$. It is easy to check that this relation is a condensation. A form of converse result also holds:

Lemma 5. *For all condensations $\sim$, there is X such that $\sim$ and $\approx_X$ coincide.*

Lemma 5 tells us that it is possible to work with condensations as if those were monadic variables. In particular, we use condensation variables in the sequel, which in fact are implemented by the set obtained from Lemma 5.

Given a convex subset I of α and some condensation $\sim$ of $\alpha|_I$, we denote by $w[I, \sim]$ the word of domain $\beta = (\alpha|_I)/\sim$ in which every $\sim$-equivalence class J is labeled by $\mathtt{value}(g, e, J)$. One can construct a formula $\mathtt{validity}(g, e)$ that checks that for all convex subsets I and all condensations $\sim$ of $\alpha|_I$ (thanks to Lemma 5), the following conditions hold:

(C1) if I is a singleton $\{x\}$, then $\mathtt{value}(g, e, I) = w(x)$,

(C2) if $w[I, \sim] = st$ for some $s, t \in S$, then $\mathtt{value}(g, e, I) = s \cdot t$,

(C3) if $w[I, \sim] = s^\omega$ for some $s \in S$, then $\mathtt{value}(g, e, I) = s^\tau$,

(C4) if $w[I, \sim] = s^{\omega^*}$ for some $s \in S$, then $\mathtt{value}(g, e, I) = s^{\tau^*}$,

(C5) if $w[I, \sim] = P^\eta$ for some $P \subseteq S$, then $\mathtt{value}(g, e, I) = P^\kappa$.

For some fixed I and $\sim$, the above tests require access to the elements $w[I, \sim](J)$ ($= \mathtt{value}(g, e, J)$), where J is a $\sim$-equivalence class of $\alpha|_I$. Since $\sim$-equivalence of two positions $x, y \in \alpha|_I$ is first-order definable (using I and $\sim$ as unary predicates), we know that for every position $x \in \alpha|_I$, the element $\mathtt{value}(g, e, [x]_\sim)$ is first-order definable in terms of x. This shows that the above properties can be expressed by first-order formulas and hence $\mathtt{validity}(g, e)$ is in the $\forall$-fragment.

The last key ingredient is to propagate those 'local validity' constraints to a 'global validity' property. This is done by the following lemma.

Lemma 6. *If* $\texttt{validity}(g, e)$ *holds, then* $\texttt{value}(g, e, I) = \pi(w|_I)$ *for all convex subsets* I *of* α*.*

This lemma implies Theorem 3. We claim indeed that, given $s \in S$, the language $\pi^{-1}(s)$ is defined by the following formula in the $\exists\forall$-fragment of MSO:

$$\texttt{evaluate}_s \ =^{\text{def}} \ \exists g.\exists e.\ \texttt{validity}(g, e) \wedge \texttt{value}(g, e, \alpha) = s \ .$$

Suppose that $\pi(w) = s$. One can find a Ramsey pair (g, e) using Theorem 4. Lemma 4 then implies $\pi(w|_I) = \texttt{value}(g, e, I)$ for all convex subsets I. Then, since π is a product, the constraints C1–C5 are satisfied and hence $\texttt{validity}(g, e)$ holds. This proves that $\texttt{evaluate}_s$ holds. Conversely, if $\texttt{evaluate}_s$ holds, this means that $\texttt{validity}(g, e)$ holds for some (g, e). In particular, Lemma 6 implies $\pi(w) = \pi(w|_\alpha) = \texttt{value}(g, e, \alpha) = s$. □

6 Conclusion

We have introduced an algebraic notion of recognizability for languages of countable words and we have shown the correspondence with the family of languages definable in MSO logic. As a byproduct of this result, we have that MSO logic interpreted over countable words collapses to its $\exists\forall$-fragment (hence, since it is closed under complementation, it also collapses to its $\forall\exists$-fragment). This collapse result is optimal, in the sense that there exist definable languages that are not definable in the $\exists$-fragment, nor in the $\forall$-fragment. An example of such a language is the set of all scattered words over $\{a\}$ and all non-scattered words over $\{b\}$: checking that a word is scattered requires a universal quantification over the sub-orderings of its domain and, conversely, checking that a word is not scattered requires an existential quantification.

References

[1] Bedon, N., Bès, A., Carton, O., Rispal, C.: Logic and rational languages of words indexed by linear orderings. Theoretical Computer Science 46(4), 737–760 (2010)

[2] Bruyère, V., Carton, O.: Automata on linear orderings. Journal of Computer and System Sciences 73(1), 1–24 (2007)

[3] Bruyère, V., Carton, O.: Automata on linear orderings. In: Proceedings of the 26th International Symposium on Mathematical Foundations of Computer Science (MFCS). LNCS, vol. 2136, pp. 236–247. Springer (2001)

[4] Büchi, J.R.: Transfinite automata recursions and weak second order theory of ordinals. In: Proc. Int. Congress Logic, Methodology, and Philosophy of Science, Jerusalem 1964. pp. 2–23. North Holland (1964)

[5] Büchi, J.: On a decision method in restricted second order arithmetic. In: Proceedings of the International Congress for Logic, Methodology and Philosophy of Science. pp. 1–11. Stanford University Press (1962)

[6] Colcombet, T.: Factorisation forests for infinite words and applications to countable scattered linear orderings. Theoretical Computer Science 411, 751–764 (2010)

[7] Feferman, S., Vaught, R.: The first-order properties of products of algebraic systems. Fundamenta Mathematicae 47, 57–103 (1959)

[8] Rabin, M.: Decidability of second-order theories and automata on infinite trees. Transactions of the American Mathematical Society 141, 1–35 (1969)
[9] Ramsey, F.: On a problem of formal logic. In: Proceedings of the London Mathematical Society. vol. 30, pp. 264–286 (1929)
[10] Rispal, C., Carton, O.: Complementation of rational sets on countable scattered linear orderings. International Journal of Foundations of Computer Science 16(4), 767–786 (2005)
[11] Rosenstein, J.: Linear Orderings. Academic Press (1982)
[12] Shelah, S.: The monadic theory of order. Annals of Mathematics 102, 379–419 (1975)
[13] Thomas, W.: Languages, automata, and logic. In: Handbook of Formal Languages, vol. 3, pp. 389–455. Springer (1997)

A Proofs for Section 3 (Semigroups and algebras for countable linear orderings)

To prove results about evaluation trees, it is convenient to first disclose a number of useful properties related to Ramsey decompositions and condensation trees.

The first two lemmas consists of (i) a well-known application of Ramsey's Theorem [9] to additive labellings and (ii) a variant of the previous result for dense orderings given by Shelah in [12]. Recall that $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$ is a finite $\circ$-algebra and, in particular, $\langle S, \cdot \rangle$ is a finite semigroup. An *additive labeling* is a function f that maps any pair of points $x < y$ in a linear ordering α to an element of the semigroup $\langle S, \cdot \rangle$ in such a way that, for every $x < y < z \in \alpha$,

$$f(x, y) \cdot f(y, z) = f(x, z).$$

Lemma 7 (Ramsey [9]). *Given a countable linear ordering α with a minimum element $\perp$ and no maximum element, and given an additive labeling $f : \alpha^2 \to \langle S, \cdot \rangle$, there exist an ω-sequence $\perp < x_1 < x_2 < ...$ of points in α and two elements $a, e \in S$ such that*

i) *for all $y \in \alpha$, there is $x_i > y$,*

ii) *$f(\perp, x_i) = a$ for all $i > 0$,*

iii) *$f(x_i, x_j) = e$ for all $j > i > 0$.*

Note that the above conditions imply that (a, e) is a *right-linked pair*, namely, $a = a \cdot e$ and $e = e \cdot e$. Moreover, it is easy to see that different right-linked pairs may result from different choices of the points $\perp < x_1 < x_2 < ...$ in α and that all such pairs are *conjugated*, namely, there exist $x, y \in S$ such that $a = b \cdot y$, $e = x \cdot y$, $b = a \cdot x$, and $f = y \cdot x$.

In the same spirit of Ramsey's Theorem, the following lemma shows that every countable dense word contains at least one perfect shuffle. Even though this result appears already in [12], we give a proof in the appendix for the sake of self-containment.

Lemma 8 (Shelah [12]). *Every word indexed by a non-singleton countable dense linear ordering contains a perfect shuffle.*

Proof. Let α be a non-singleton countable dense linear ordering, $A = \{a_1, ..., a_n\}$ a generic alphabet, and w a word over A of domain α. For the sake of brevity, given a symbol a, we denote by $w^{-1}(a)$ the set of all points $x \in \alpha$ such that $w(x) = a$. We then define $w_0 = w$ and $A_0 = \emptyset$ and we recursively apply the following construction for each index $1 \leq i \leq n$:

$$A_i = \begin{cases} A_{i-1} \cup \{a_i\} & \text{if } w^{-1}(a_i) \text{ is dense in } \alpha, \\ A_{i-1} & \text{otherwise,} \end{cases}$$

$$w_i = \begin{cases} w_{i-1} & \text{if } w^{-1}(a_i) \text{ is dense in } \alpha, \\ w_{i-1}|_I & \begin{array}{l}\text{if there is an open non-empty non-singleton} \\ \quad \text{convex subset } I \text{ of } \alpha \text{ such that } w^{-1}(a_i) \cap I = \emptyset.\end{array} \end{cases}$$

Clearly, the sub-word w_n is a perfect shuffle of the set A_n. □

We now turn to the properties of condensation trees. The following lemma, whose proof is trivial and thus omitted, shows how it is possible to restrict a condensation tree to any convex subset.

Lemma 9. *Given a condensation tree T over a linear ordering α and a convex subset I of α, define the* subtree of T rooted at I *as follows:*

$$T|_I \ = \ \{I \cap J \ : \ J \in T, \ I \cap J \neq \emptyset\}.$$

Then $T|_I$ is a condensation tree over $\alpha \cap I$. Furthermore, for all convex subsets J such that $I \subseteq J \subseteq \alpha$, we have $(T|_I)|_J = T|_{I \cap J}$.

We will often perform inductions on trees, where the parameter, called *rank*, is introduced in the next lemma. At the same time, we show how the notion of restriction of a condensation tree is compatible with that of rank.

Lemma 10. *It is possible to associate with each condensation tree T an ordinal $\mathtt{rank}(T)$ such that, for all condensation trees T over a linear ordering α and all convex subsets I of α, $\mathtt{rank}(T|_I) \leq \mathtt{rank}(T)$. Furthermore, if I is included in some node $J \in T$ that is not the root of T, then $\mathtt{rank}(T|_I) < \mathtt{rank}(T)$.*

Proof. Let T be a condensation tree over a linear ordering α. We associate with each node I of T a suitable countable ordinal γ_I as follows. For every leaf I of T, we simply let $\gamma_I = 1$. Then, given an internal node I of T, we assume that γ_J is defined for every child J of I in T and we define γ_I as the ordinal $\{\gamma \ : \ \exists\, J \in \mathtt{children}(I).\ \gamma \leq \gamma_J\}$ (note that this is either a successor ordinal or a limit ordinal, depending on whether the set $\{\gamma_J \ : \ J \in \mathtt{children}(I)\}$ has a maximum element or not). Since T has no infinite branch, it follows that γ_I is defined for every node of T. We thus let $\mathtt{rank}(T) = \gamma_I$, where I is the root of T. It is easy to check that the function $\mathtt{rank}$ that maps any condensation tree T to its rank $\mathtt{rank}(T)$ satisfies the properties stated in the lemma. □

Each time we perform an 'induction' on a condensation tree, this means that the proof is in fact by induction on its rank. Note that the validity of the induction is justified by Lemma 10 (we will not recall this in the proofs).

We now focus on evaluation trees. The following lemma shows that if the partial mapping π_0 is defined over a word, then it is also defined over its factors. We further allow some change of values at the extremities of words and make some case distinctions for dealing with the empty word. This makes the statement a bit more technical (the proof by case distinction is straightforward and thus omitted):

Lemma 11. *Given two (possibly empty) words u and v and an element $c \in S$, if $\pi_0(ucv)$ is defined, then $\pi_0(ua)$ and $\pi_0(bv)$ are also defined for all elements $a, b, c \in S \cup \{\varepsilon\}$ for which $a = b = \varepsilon$ implies $c = \varepsilon$. Furthermore, if $c = a \cdot b$, then $\pi_0(ucv) = \pi_0(ua) \cdot \pi_0(bv)$.*

Next, we prove that given an evaluation tree over some word, there exist evaluation trees over all its factors:

Lemma 12. *For every evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over a word u of domain α and every convex subset I of α, there is an evaluation tree $\mathcal{T}|_I = \langle T|_I, \gamma_I \rangle$ such that γ_I and γ coincide over $(T|_I) \cap T = \{J \in T \ : \ J \subseteq I\}$.*

Proof. Let us first assume that I is an initial segment of α, namely, for every $y \in I$ and every $x \leq y$, $x \in I$. The proof is by induction on $\mathcal{T}$, namely, on the rank of the underlying condensation tree T. Let C be the top-level condensation $\texttt{children}(\alpha)$. We distinguish between two sub-cases.

If the condensation $\{I, \alpha \setminus I\}$ is coarser than C, then for all $K \in T|_I$ with $K \neq I$, we have $K \in T$. Hence it makes sense to define $\gamma_I(K) = \gamma(K)$. We complete the definition by letting $\gamma_I(\alpha) = \pi_0(\gamma(C|_I))$, where $\gamma(C|_I)$ is the word of domain $C|_I = \{K \cap I \ : \ K \in C\}$ where each position J is labeled by the value $\gamma(J)$ (note that, by Lemma 11, the function π_0 is defined on the word $\gamma(C|_I)$). It is easy to check that $\langle T|_I, \gamma_I \rangle$ thus defined is an evaluation tree over $u|_I$ and that γ_I and γ coincide over $(T|_I) \cap T = \{K \in T \ : \ K \subseteq I\}$.

Otherwise, there exist three convex subsets $J_1 < J_2 < J_3$ of α such that (i) $\{J_1, J_2, J_3\}$ forms a partition coarser than C, (ii) $J_1 \subseteq I$, (iii) $J_3 \cap I = \emptyset$, and (iv) $J_2 \in C$ with $J_2 \cap I \neq \emptyset$ and $J_2 \setminus I \neq \emptyset$. We now apply the induction hypothesis to construct the evaluation tree $\mathcal{T}|_{J_2 \cap I} = \langle T|_{J_2 \cap I}, \gamma_{J_2 \cap I} \rangle$. Note that for every $K \in T|_{J_1}$ with $K \neq J_1$, we have $K \in T$. Hence it makes sense to define $\gamma_I(K) = \gamma(K)$. For every $K \in T|_{J_2}$, we define $\gamma_I(K) = \gamma_{J_2 \cap I}(K)$. Finally, we define $\gamma_I(\alpha) = \pi_0\big(\gamma_I(C|_{J_1})\, \gamma_{J_2 \cap I}(J_2 \cap I)\big)$ (again this is well-defined according to Lemma 11). It is easy to check that $\langle T|_I, \gamma_I \rangle$ thus defined is an evaluation tree over $u|_I$ and that γ_I and γ coincide over $(T|_I) \cap T = \{K \in T \ : \ K \subseteq I\}$.

Finally, we consider the case where I is not an initial segment of α. In this case it is possible to write I as $I_1 \cap I_2$, where I_1 is an initial segment and I_2 is a final segment of α. Since $T|_I = (T|_{I_1})|_{I_2}$, it is sufficient to apply twice the cases for the initial/final segment discussed above. $\square$

The consequence of the above lemma is that an evaluation tree does not only provide a value for a word, but also, by restriction, for all its factors. This is concretized in the following notation: given an evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over a word u of domain α and given a convex subset I of α, we denote by $\gamma(I)$ the value associated with I in the evaluation tree $\mathcal{T}|_I$ (note that the notation is consistent with the value associated with I in the evaluation tree $\mathcal{T}$, when $I \in T$). Intuitively, this means that the mapping γ of $\mathcal{T}$ can be prolonged to all convex subsets of α.

We now turn to the proofs of Proposition 1 and Proposition 2.

Proposition 1. *For every word u, there exists an evaluation tree over u.*

Proof. Let u be a word of domain α. We say that a convex subset I of α is *definable* if there is an evaluation tree over the factor $u|_I$. Observe that, in virtue of Lemma 12, if I is definable, then all its convex subsets J are definable as well.

We first establish the following claim: for every ascending chain $I_1 \subseteq I_2 \subseteq ...$ of definable convex subsets of α, the limit $I = \bigcup_{i\in\omega} I_i$ is definable. Of course, if the sequence of the I_i's is ultimately constant, then the claim holds trivially. Otherwise, let us consider first the case when all the I_i's coincide on the left. Clearly, we can partition I into a sequence of non-empty convex subsets $J_0 < J_1 < ...$, forming a condensation of I, such that $I_i = J_1 \cup ... \cup J_i$ for all $i \in \omega$. For every $i < j$ in ω, we define $J_{i,j} = J_i \cup ... \cup J_{j-1}$. We recall that every set I_j, and hence every convex subset $J_{i,j}$ of it, is definable. We can thus associate with each convex subset $J_{i,j}$ an evaluation tree $\mathcal{T}_{i,j}$ over $u|_{J_{i,j}}$. We denote by $c_{i,j}$ the value associated with $J_{i,j}$ in the evaluation tree $\mathcal{T}_{i,j}$. Using Lemma 7 (i.e., Ramsey's Theorem), one can extract a sequence $0 < i_1 < i_2 < ...$ in ω such that $c_{i_1,i_2} = c_{i_2,i_3} = ...$ is an idempotent. We can then construct an evaluation tree over $u|_J$ that has root I and the convex subsets J_{0,i_1}, J_{i_1,i_2}, ... for children, with the associated evaluation subtrees $\mathcal{T}_{0,i_1}$, $\mathcal{T}_{i_1,i_2}$, ... This proves that I is definable. The case where the convex the sets I_i coincide on the right is symmetric. Finally, in the general case, we can partition each set I_i into two subsets I'_i and I''_i such that (i) $I'_i < I''_i$, (ii) the sequence of the I'_i's coincide on the right, and (iii) the sequence of the I''_i's coincide on the left. Let $I' = \bigcup_{i\in\omega} I'_i$ and $I'' = \bigcup_{i\in\omega} I''_i$. One knows by the cases above that both sets I' and I'' are definable. At this point, one easily provides an evaluation tree for $u|_I = u|_{I'\cup I''}$ out of the evaluation trees for I' and I''. Hence I is definable. This concludes the proof of the claim.

Let us now consider the set $\mathcal{C}$ of all condensations C of α such that every class is definable. Condensations in $\mathcal{C}$ are naturally ordered by the 'coarser than' relation. Let us consider a chain $(C_i)_{i\in\beta}$ of condensations in $\mathcal{C}$ ordered by the coarser than relation. Since α is countable, one can assume that β is countable, or even better that $\beta = \omega$. Let us consider the limit condensation C. Each class $I \in C$ is limit of a sequence of convex subsets I_i, with $I_i \in C_i$ for all $i \in \omega$. From the assumption that each condensation C_i belongs to $\mathcal{C}$, we get that I_i is definable and from the claim above, we conclude that I is definable as well. This shows that the condensation C belongs to $\mathcal{C}$ and hence every chain of $\mathcal{C}$ has a least upper bound in $\mathcal{C}$.

It follows that we can apply Zorn's Lemma and deduce that $\mathcal{C}$ contains a maximal element, say C. If C has a single equivalence class, this means that there exists an evaluation tree over u and the proposition is established. Otherwise, we shall head toward a contradiction. Consider the condensed linear ordering induced by C (by a slight abuse of notation, we denote it also by C). Two cases can happen: either C contains two consecutive classes or C is a linear order dense in itself. In the former case, we fix two consecutive classes $I, I' \in C$, with $I < I'$, and we denote by $\mathcal{T}$ and $\mathcal{T}'$ some evaluation trees over $u|_I$ and $u|_{I'}$, respectively (note that such evaluation trees exist because each class of C is a limit of definable convex subsets and hence, by the previous claim, it is definable as well). It is easy to construct an evaluation tree over $u|_{I\cup I'}$, where the root $I \cup I'$ has $\mathcal{T}|_I$ and $\mathcal{T}|_{I'}$ as direct subtrees. This would imply that the convex subset

$I \cup I'$ is definable, which is against the hypothesis. In the second case, the linear ordering C is dense in itself. We associate with each class $I \in C$ an evaluation tree $\mathcal{T}_I$ over $u|_I$ and we denote by c_I the value associated with I in it. Let us now consider the word $v = \Pi_{I \in C} c_I$. We know from Lemma 8 that that v contains a perfect shuffle v' of domain $C' \subseteq C$. Let $J = \bigcup_{I \in C'} I$. We can easily construct an evaluation tree $\mathcal{T}$ over $u|_J$, whose root is J and whose direct subtrees are the evaluation trees $\mathcal{T}_I$ associated with each class $I \in C'$. This would imply that the convex subset J is definable, which is against the hypothesis. $\square$

Proposition 2. *Two evaluation trees over the same word have same value.*

The proof of this result is more technical. Our first step is a key lemma which explains why the axioms for the $\circ$-algebras have been introduced. This is a form of associativity rule for π_0, but such that the equality is required to hold solely when every expression is defined.

Lemma 13. *For every word u of the form $\Pi_{i \in \alpha} u_i$, where α is a countable linear ordering and each u_i is a word in S°, if both $\pi_0(u)$ and $\pi_0\big(\Pi_{i \in \alpha} \pi_0(u_i)\big)$ are defined, then the two values are equal.*

Proof. We prove the claim by case analysis. For the sake of brevity, we define $v = \Pi_{i \in \alpha} \pi_0(u_i)$. If $u = a_1 ... a_n$, then v has to be of the form $b_1 ... b_m$. Since $\cdot$ is associative (see Axiom A1), we obtain $\pi_0(u) = a_1 \cdot ... \cdot a_n = b_1 \cdot ... \cdot b_m = \pi_0(v)$. If $u = ae^\omega$, then v can be either of the form $b_1 ... b_m$ or of the form bf^ω. If $v = b_1 ... b_m$, say with $m \geq 2$ (the case $n = 1$ is obvious), then we necessarily have $b_1 \in \{a, a \cdot e\}$, $b_i = e$ for all $1 < i < m$, and $b_m = e^\tau$. Moreover, Axioms A1 and A2 together imply $e \cdot e^\tau = e \cdot (e \cdot e)^\tau = (e \cdot e)^\tau = e^\tau$. We thus have $\pi_0(u) = a \cdot e^\tau = b_1 \cdot ... \cdot b_m = \pi_0(v)$. If $v = bf^\omega$, then, as above, we get $b \in \{a, a \cdot e\}$ and $f = e$. Using Axioms A1 and A2 one can easily prove $\pi_0(u) = a \cdot e^\tau = b \cdot f^\tau = \pi_0(v)$. The case $u = e^{\omega^*} a$ is symmetric and it uses Axiom A3 instead of Axiom A2.

Finally, the most interesting case is when $u = a \cdot P^\eta \cdot a'$ for some $a, a' \in S \cup \{\varepsilon\}$ and some non-empty set $P \subseteq S$. We further distinguish some cases depending on the form of v.

- If $v = b_1 ... b_m$, then the proof goes by induction on m. The interesting case is $m = 2$. where five sub-cases can occur. If u_1 has no last letter and u_2 has no first letter, then we have $b_1 = \pi_0(u_1) = a \cdot P^\kappa$ and $b_2 = \pi_0(u_2) = P^\kappa \cdot a'$. Using Axiom A4, we get $\pi_0(u) = a \cdot P^\kappa \cdot a' = (a \cdot P^\kappa) \cdot (P^\kappa \cdot a') = b_1 \cdot b_2 = \pi_0(v)$. If u_1 consists of a single letter, then this letter must be $a \neq \varepsilon$ and u_2 has no first letter. Hence, as above, we have $\pi_0(u_2) = P^\kappa \cdot a'$. We thus get $\pi_0(u) = a \cdot P^\kappa \cdot a' = b_1 \cdot b_2 = \pi_0(v)$. If u_1 has a last letter c and at least two letters, then c must belong to P and u_2 has no first letter. We thus have $\pi_0(u) = a \cdot P^\kappa \cdot a' = (a \cdot P^\kappa \cdot c) \cdot (P^\kappa \cdot a') = \pi_0(v)$. The cases where u_2 has length 1 and where u_2 has a first letter and at least two letters are symmetric. Finally, the induction for $m > 2$ is straightforward.
- If $v = bf^\omega$, then, by distinguishing some sub-cases as above, one verifies that $b = \pi_0(u_1)$ coincides with either a or $a \cdot P^\kappa \cdot c$, for some $c \in P \cup \{\varepsilon\}$, and that $f = \pi_0(u_2) = \pi_0(u_3) = ...$ coincides with either $P^\kappa \cdot d$ or $d \cdot P^\kappa$,

for some $d \in P \cup \{\varepsilon\}$, depending on whether u_1 has a first letter or not. Putting all together and using Axiom A4, we have either $\pi_0(u) = a \cdot P^\kappa = a \cdot (P^\kappa \cdot d)^\tau = \pi_0(v)$, or $\pi_0(u) = a \cdot P^\kappa = (a \cdot P^\kappa \cdot c) \cdot (P^\kappa \cdot d)^\tau = \pi_0(v)$, or $\pi_0(u) = a \cdot P^\kappa = (a \cdot P^\kappa) \cdot (d \cdot P^\kappa) = \pi_0(v)$.

- If $v = f^{\omega^*}b$, then the claim holds by symmetry with the previous case.
- If $v = bR^\eta b'$ for some $b, b' \in S \cup \{\varepsilon\}$ and some non-empty set $R \subseteq S$, then we prove that R is included in $P \ \cup \ (P \cup \{\varepsilon\}) \cdot P^\kappa \cdot (P \cup \{\varepsilon\})$. Let us treat first the case $b = b' = \varepsilon$. Since v has no first nor final letter, this implies $a = a' = \varepsilon$. Let us consider a value $c \in R$ and a corresponding factor u_i of u, with $i \in \alpha$, such that $\pi_0(u_i) = u_i$. If u_i consists of the single letter c, then we clearly have $c \in P$. Otherwise u_i has more than one letter and we get the four possibilities $c = P^\kappa$, $c = d{\cdot}P^\kappa$, $c = P^\kappa{\cdot}d'$ and $c = d{\cdot}P^\kappa{\cdot}d'$, for suitable $d, d' \in P$, depending on the existence of a first/last letter in u_i. This proves that R is included in $P \ \cup \ (P \cup \{\varepsilon\}) \cdot P^\kappa \cdot (P \cup \{\varepsilon\})$. Using Axiom A4 we immediately obtain $\pi_0(u) = P^\kappa = R^\kappa = \pi_0(v)$. The general case where $b, b' \in S \cup \{\varepsilon\}$, can be dealt with by using similar arguments plus Axiom A1. □

Corollary 1. *Let u be a word of domain α such that $\pi_0(u)$ is defined and let $\mathcal{T} = \langle T, \gamma \rangle$ be an evaluation tree over u. Then $\pi_0(u) = \gamma(\alpha)$.*

Proof. We prove the claim by induction on $\mathcal{T}$. If $\mathcal{T}$ consists of a single leaf, then the claim is obvious. Otherwise, let $C = \texttt{children}(\alpha)$. By Lemma 11, we know that for every $I \in C$, $\pi_0(u|_I)$ is defined. We can then use the induction hypothesis on the evaluation tree $\mathcal{T}|_I$ and obtain $\pi_0(u|_I) = \gamma(I)$. Finally, using Lemma 13, we get $\pi_0(u) = \pi_0(\gamma(C)) = \gamma(\alpha)$. □

The next step is to prove the equality between the value at the root of an evaluation tree and the values induced by π_0 under different condensations of the root. We first consider finite condensations, then ω-condensations (and, by symmetry, ω^*-condensations), and finally η-condensations. For a technical reason, we have to deal with the cases where additional symbols appear at the beginning and at the end of a word. The gathering of those results naturally entail that two evaluation trees over the same word have the same value (see Corollary 3).

Lemma 14. *Given a word u of domain α, an evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over u, and a finite condensation $I_1 < ... < I_n$ of α, we have $\gamma(\alpha) = \gamma(I_1) \cdot ... \cdot \gamma(I_n)$.*

Proof. The proof is by induction on $\mathcal{T}$. If $\mathcal{T}$ consists of a single node, then this node must be a leaf and α must be a singleton. Therefore, we have $n = 1$ and the claim follows trivially. Let us now consider the case where $\mathcal{T}$ has more than one node. We only prove the claim for $n = 2$ (for $n = 1$ it is obvious and for $n > 2$ it follows from a simple induction). Let C be the condensation $\texttt{children}(\alpha)$ and let J be the unique convex subset in C that intersects both I_1 and I_2 (if C does not contain such an element, then we let $J = \emptyset$). For the sake of brevity, we define, for both $i = 1$ and $i = 2$, $C_i = \{K \in C \ : \ K \subseteq I_i\}$,

$u_i = \prod_{K \in C_i} \gamma(K)$, and $a_i = \gamma(J \cap I_i)$ (with the convention that $\gamma(J \cap I_i) = \varepsilon$ if $J = \emptyset$). Note that $C = C_1 \cup \{J\} \cup C_2$ and hence $\gamma(\alpha) = \pi_0(u_1\gamma(J)u_2)$. Moreover, if J is not empty, then, by applying the induction hypothesis to the evaluation tree $\mathcal{T}|_J$ and the condensation $\{J \cap I_1, J \cap I_2\}$ of $\alpha|_J$, we obtain $\gamma(J) = \gamma(J \cap I_1) \cdot \gamma(J \cap I_2) = a_1 \cdot a_2$ and hence $\gamma(\alpha) = \pi_0(u_1\ (a_1 \cdot a_2)\ u_2)$. Lemma 11 then implies $\pi_0(u_1\ (a_1 \cdot a_2)\ u_2) = \pi_0(u_1 a_1) \cdot \pi_0(u_2 a_2)$. Similarly, Lemma 13 implies $\pi_0(u_1 a_1) = \gamma(I_1)$ and $\pi_0(u_2 a_2) = \gamma(I_2)$. Overall, we get $\pi_0(\alpha) = \gamma(I_1) \cdot \gamma(I_2)$. □

Lemma 15. *Given a word u of domain α, an evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over u, and an ω-condensation $I_0 < I_1 < I_2 < ...$ of α such that $\gamma(I_1) = \gamma(I_2) = ...$ is an idempotent, we have $\gamma(\alpha) = \gamma(I_0) \cdot \gamma(I_1)^\tau$.*

Proof. The proof is by induction on $\mathcal{T}$. Note that the case of $\mathcal{T}$ consisting of a single leaf cannot happen. Let C be the condensation $\texttt{children}_t(\alpha)$. We distinguish two cases depending on whether C has a maximal element or not.

If C has a maximal element, say J_{max}, then we can find a condensation $K_1 < K_2$ of α that is coarser than $I_0 < I_1 < I_2 < ...$ and such that $K_2 \subseteq J_{max}$. By Lemma 14, we have $\gamma(\alpha) = \gamma(K_1) \cdot \gamma(K_2)$. Moreover, since K_1 is the union of a finite sequence of convex subsets $I_0, I_1, ..., I_k$, by repeatedly applying Lemma 13, we obtain $\gamma(K_1) = \gamma(I_0) \cdot \gamma(I_1) \cdot ... \cdot \gamma(I_k) = \gamma(I_0) \cdot \gamma(I_1)$ (the last equality follows from the fact that $\gamma(I_1) = \gamma(I_2) = ...$ is an idempotent). Finally, from the induction hypothesis, we get $\gamma(K_2) = \gamma(I_1)^\tau$. We thus conclude that $\gamma(\alpha) = \big(\gamma(I_0) \cdot \gamma(I_1)\big) \cdot \big(\gamma(I_1)^\tau\big) = \gamma(I_0) \cdot \gamma(I_1)^\tau$.

If C has no maximal element, then, using standard extraction techniques and Ramsey's Theorem (though it is not really necessary), one can construct an ω-condensation $J_0 < K_1 < J_1 < K_2 < J_2 < ...$ of α such that:

- $\{J_0 \cup K_1, J_1 \cup K_2, ...\}$ is coarser than $\{I_0, I_1, I_2, ...\}$,
- $\{J_0, K_1 \cup J_1, K_2 \cup J_2, ...\}$ is coarser than C,
- $\gamma(K_1 \cup J_1) = \gamma(K_2 \cup J_2) = ...$ is an idempotent.

Let $\gamma(C)$ be the word of domain C where each position $H \in C$ is labeled by the value $\gamma(H)$. By construction, we have $\gamma(\alpha) = \pi_0(\gamma(C))$. Moreover, since the condensation $\{J_0, K_1 \cup J_1, K_2 \cup J_2, ...\}$ is coarser than C, by repeatedly applying Lemma 13, we obtain $\pi_0(\gamma(C)) = \pi_0\big(\gamma(J_0)\ \gamma(K_1 \cup J_1)\ \gamma(K_2 \cup J_2)\ \dots\big) = \gamma(J_0) \cdot \gamma(K_1 \cup J_1)^\tau$. Similarly, since $\{J_0 \cup K_1, J_1 \cup K_2, ...\}$ is coarser than $\{I_0, I_1, I_2, ...\}$ and $\gamma(I_1) = \gamma(I_2) = ...$ is an idempotent, we have $\gamma(J_0 \cup K_1) = \gamma(I_0) \cdot \gamma(I_1)$ and $\gamma(J_1 \cup K_2) = \gamma(J_2 \cup K_3) = ... = \gamma(I_1)$. Thus, by Axioms A1 and A2, we obtain $\gamma(J_0) \cdot \gamma(K_1 \cup J_1)^\tau = \gamma(I_0) \cdot \gamma(I_1)^\tau$. □

We can gather all the results seen so far and prove the following corollary.

Corollary 2. *Given a word u of domain α, an evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over u, a scattered condensation C of α, and an evaluation tree $\mathcal{T}' = \langle T', \gamma' \rangle$ over the word $\gamma(C) = \prod_{I \in C} \gamma(I)$ of domain C, we have $\gamma(\alpha) = \gamma'(C)$.*

Proof. As a preliminary remark, note that since the condensation C is scattered, we have that, for every node I' in the evaluation tree $\mathcal{T}' = \langle T', \gamma' \rangle$, the condensation of I' induced by $\mathcal{T}'$ is scattered as well. The proof is by induction on $\mathcal{T}'$.

If $\mathcal{T}'$ consists of a single node, then $\gamma(C)$ is a singleton word of value $\gamma(\alpha)$ and hence the statement boils down to $\gamma(\alpha) = \gamma(\alpha)$. Otherwise, let C' be the set of children of the root C in $\mathcal{T}'$. From the induction hypothesis, we know that for every $I' \in C'$, $\gamma'(I') = \gamma\big(\bigcup I'\big)$, where $\bigcup I'$ denotes the union of all convex subsets of I' (recall that $I' \subseteq C$). Moreover, if we denote by $\bigcup C'$ the condensation of α obtained from the substitution of each element $I' \in C'$ by $\bigcup I'$, we have

$$\gamma'(C) \;=\; \pi_0\left(\prod_{I' \in C'} \gamma'(I')\right) \;=\; \pi_0\left(\prod_{I' \in C'} \gamma\big(\bigcup I'\big)\right) \;=\; \pi_0\left(\gamma\big(\bigcup C'\big)\right).$$

Note that the condensation $\bigcup C'$ of α as the same order-type of the condensation C' of C, namely, it is either a finite condensation, an ω-condensation, or an ω^*-condensation. Therefore, using either Lemma 14 or Lemma 15 (or its symmetric variant), we obtain $\pi_0(\gamma(\bigcup C')) = \gamma(\alpha)$. □

Lemma 16. *Given a word u of domain α, an evaluation tree $\mathcal{T} = \langle T, \gamma\rangle$ over u, and a dense condensation C of α such that $\gamma(C) = \prod_{I \in C} \gamma(I)$ is isomorphic to $aP^\eta b$, for some elements $a, b \in S \cup \{\varepsilon\}$ and some non-empty set $P \subseteq S$, we have $\gamma(\alpha) = a \cdot P^\kappa \cdot b$.*

Proof. We remark here that the proof works for any condensation C, independently of the form of the word $\gamma(C)$; however, the use of the following technical arguments does only make sense when C is a dense condensation. We prove the lemma by induction on $\mathcal{T}$. As in the proof of Lemma 15, the case of $\mathcal{T}$ consisting of a single node cannot happen. Let D be the condensation $\texttt{children}(\alpha)$ and let E be the finest condensation that is coarser than or equal to both C and D (note that E exists since condensations form a lattice structure with respect to the 'coarser-than' relation). Moreover, let $\sim$ be the condensation over C such that, for every $I, I' \in C$, $I \sim I'$ holds iff either $I = I'$ or there is $J \in D$ with $I \subseteq J$ and $I' \subseteq J$. This can naturally be seen as a condensation C' over α which is at least as coarse as C: precisely, the classes of C' are either the single classes of C that are not contained in any class of D, or the unions of the classes of C that are contained in the same class of D. Furthermore, it is easy to see that E is at least as coarse as C'. We start by disclosing some properties of the condensations C, D, E, and C'.

Let us first consider a class $I \in C'$. Two cases can happen: either I is included in some $J \in D$, and in this case $\gamma(I) = \pi_0(\gamma(C|_I))$ holds by induction hypothesis, or I belongs to C, and hence $\gamma(I) = \pi_0(\gamma(C|_I))$ follows immediately. We have just proved that

$$\forall\, I \in C'. \quad \gamma(I) \;=\; \pi_0(\gamma(C|_I)). \tag{1}$$

Now, let I, I' be two distinct classes in C'. We claim that there exist $x \in I$ and $x' \in I'$ that are not equivalent for D, namely,

$$\exists\, x \in I.\ \exists\, x' \in I'.\ \forall\, J \in D. \quad x \notin J \;\vee\; x' \notin J. \tag{2}$$

The proof of this property is by case distinction. If I is contained in some $J \in D$ and I' is contained in some $J' \in D$, then we necessarily have $J \neq J'$ (otherwise, we would have $I = I'$ by definition of C') and hence Property (2) holds. Otherwise, either I is not contained in any class $J \in D$, or I' is not contained in any class $J \in D$. Without loss of generality, we assume that I is not contained in any class $J \in D$. This means that there exists $J \in D$ such that $I \cap J \neq \emptyset$ and $I \setminus J \neq \emptyset$. Let us pick some $x' \in I'$. Clearly, x' belongs to some $J' \in D$. Then either $J \cap J' = \emptyset$ or $J = J'$. In the first case, one chooses $x \in I \cap J$, while in the second case one chooses $x \in I \setminus J$. This completes the proof of Property (2).

From the above property, we can deduce the following:

$$\begin{array}{l}\textit{If } I, I' \in C',\ I < I',\ \textit{and } I, I' \subseteq K \textit{ for some } K \in E, \textit{ then} \\ \textit{there are only finitely many classes } I'' \in C' \textit{ between } I \textit{ and } I'.\end{array} \tag{3}$$

Indeed, suppose that the above property does not hold, namely, that there are infinitely many classes $I'' \in C'$ between I and I'. In particular, we can find an ω-sequence of classes $I_1, I_2, ...$ such that $I = I_1 < I_2 < ... < I'$ or $I < ... < I_2 < I_1 = I'$. We only consider the first case (the second case is symmetric). By applying Property (2) to the classes $I_1, I_2, ...$, we can find some points $x_1 \in I_1$, $x'_1 \in I_2$, $x_2 \in I_3$, $x'_2 \in I_4$, ... such that, for all $i \in \omega$, x_i and x'_i are not equivalent for D (i.e., for all $J \in D$, $x_i \notin J$ or $x'_i \notin J$). Let X be the set of all points $x \in \alpha$, with $x < I_i$ for some $i \in \omega$, and let X' be the set of all points $x' \in \alpha$, with $x' > I_j$ for all $j \in \omega$. Since D is a condensation, we have that for all $x \in X$ and all $x' \in X'$, x and x' are not equivalent for D. Moreover, by construction, all such points x and x' are not equivalent for C', and hence for C either (recall that C is finer than C'). Since E is the defined as the finest condensation that is coarser than or equal to both C and D and since $X \cup X' = \alpha$, it follows that there is no class $K \in E$ that intersects both X and X'. In particular, since $I \subseteq X$ and $I' \subseteq X'$, it follows that there is no class $K \in E$ such that $I \subseteq K$ and $I' \subseteq K$, which is a contradiction. This completes the proof of Property (3).

We prove the following last property:

$$\forall\, K \in E. \quad \gamma(K) \;=\; \pi_0(\gamma(C|_K)). \tag{4}$$

Let $K \in E$ and let $\mathcal{T}' = \langle T', \gamma' \rangle$ be an evaluation tree over the word $\gamma(C'|_K)$ (such a tree exists according to Proposition 1). From Property (3) we know that the condensation of $C'|_K$ induced by the evaluation tree $\mathcal{T}'$ is scattered. We can thus apply Corollary 2 and obtain $\gamma(K) = \gamma'(C'|_K)$. Moreover, the value $\pi_0(\gamma(C'|_K))$ is defined and hence, by Corollary 1, $\gamma'(C'|_K) = \pi_0(\gamma(C'|_K))$. By Property (1), we obtain $\gamma(C'|_K) = \prod_{I \in C'|_K} \gamma(I) = \prod_{I \in C'|_K} \pi_0(\gamma(C|_I))$. Finally, from the properties of condensation trees, we derive $\pi_0(\gamma(C'|_K)) = \pi_0\big(\prod_{I \in C'|_K} \pi_0(\gamma(C|_I))\big) = \gamma(C|_K)$. This completes the proof of Property (4).

Towards a conclusion, we consider an evaluation tree $\mathcal{T}'' = \langle T'', \gamma'' \rangle$ over the word $\gamma(E)$ (such a tree exists according to Proposition 1). From Property (4) we know that $\gamma(E) = \prod_{K \in E} \gamma(K) = \prod_{K \in E} \pi_0(\gamma(C|_K))$. Moreover, By Corollary 1, we know that $\pi_0(\gamma(C|_K)) = \gamma''(K)$ and hence $\prod_{K \in E} \pi_0(\gamma(C|_K)) = \gamma''(E)$. Similarly, since E is at least as coarse as D, Corollary 1 implies $\gamma''(E) = \pi_0(\gamma(D)) = \gamma(\alpha)$. This completes the proof of the lemma. $\square$

Corollary 3. *Given a word u of domain α, an evaluation tree $\mathcal{T} = \langle T, \gamma \rangle$ over u, a condensation C of α, and an evaluation tree $\mathcal{T}' = \langle T', \gamma' \rangle$ over the word $\gamma(C) = \prod_{I \in C} \gamma(I)$ of domain C, we have $\gamma(\alpha) = \gamma'(C)$.*

Proof. The proof is exactly the same as for Corollary 2, with the only difference that we do not use the assumption that the condensation C is scattered and we use Lemma 16 for treating the nodes I' of $\mathcal{T}'$ for which the condensation $\mathtt{children}(I')$ is dense. □

Finally, the claim of Proposition 2 follows easily from the previous corollary.

Proof (of Proposition 2). Let $\mathcal{T} = \langle T, \gamma \rangle$ and $\mathcal{T}' = \langle T', \gamma' \rangle$ be two evaluation trees over the same word u of domain α and let C be the minimal condensation of α, whose classes are the singleton sets. Clearly, the evaluation tree $\mathcal{T}'$ is isomorphic to an evaluation tree $\mathcal{T}'' = \langle T'', \gamma'' \rangle$ over the word $\gamma(C) = \prod_{I \in C} \gamma(I)$ of domain C. Using Corollary 3 we immediately obtain that $\gamma(\alpha) = \gamma''(C) = \gamma'(\alpha)$.

We conclude this part with an example of a $\circ$-semigroup induced by a finite $\circ$-algebra.

Example 1. Consider a $\circ$-algebra $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$ that consists of exactly five elements, a, b, c, d, and $\perp$, and whose operators satisfy, besides Axioms A1-A4, the following rules:

	$\cdot a$	$\cdot b$	$\cdot c$	$\cdot d$	$\cdot \perp$	τ	τ^*	κ
a	a	a	c	c	$\perp$	a	a	a
b	b	b	d	d	$\perp$	b	a	a
c	a	a	c	c	$\perp$	a	c	a
d	b	b	d	d	$\perp$	b	c	$\perp$
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

Note that, in this specific case, the value P^κ, for any non-empty subset $P \subseteq S$, is uniquely determined by the above rules (for instance, by Axiom 4, $\{b, c\}^\kappa = s$ implies $\{b \cdot s \cdot c\}^\kappa = s$, whence $s = \perp$, whence $\{b, c\}^\kappa = \perp$). This basically means that the above table uniquely determines a $\circ$-algebra $\langle S, \cdot, \tau, \tau^*, \kappa \rangle$. Now, by Theorem 1, such an algebra induces a corresponding $\circ$-semigroup $\langle S, \pi \rangle$ such that, for every word w over S:

- $\pi(w) = a$ iff (i) the first symbol of w (if any) is a, (ii) the last symbol of w (if any) is a, and (iii) w does not contain any sub-word u such that $\pi(u) = \perp$;
- $\pi(w) = b$ iff (i) w begins with either b or d, (ii) the last symbol of w (if any) is either a or b, and (iii) w does not contain any sub-word u such that $\pi(u) = \perp$;
- $\pi(w) = c$ iff (i) the first symbol of w (if any) is either a or c, (ii) w ends with either c or d, and (iii) w does not contain any sub-word u such that $\pi(u) = \perp$;

- $\pi(w) = d$ iff (i) w begins with either b or d, (ii) w ends with either c or d, and (iii) w does not contain any sub-word u such that $\pi(u) = \perp$;
- $\pi(w) = \perp$ iff w contains at least one occurrence of the symbol $\perp$ or it can be written as a product of the form $\prod_{i \in \alpha} u_i$ such that the word $v = \prod_{i \in \alpha} \pi(u_i)$ contains a perfect shuffle of some set $P \supseteq \{d\}$ or $P \supseteq \{b, c\}$.

B Proofs for Section 5 (From ○-algebras to monadic second-order logic)

In this section, when we say that a monadic variable I is first-order definable from other variables $\overline{X}$, this means that there exists a first order formula $\phi(x,\overline{X})$ which holds iff $x \in I$. In practice, this means that it is never necessary to quantify over I for defining properties concerning I. It is sufficient to replace each predicate $x \in I$ by the corresponding formula $\phi(x,\overline{X})$. This remark is necessary for understanding why the construction we provide yields a formula in the $\exists\forall$-fragment of MSO.

Lemma 4. *For all $s \in S$, there is a first-order formula $\mathtt{value}_s(g,e,X)$, such that for every convex subset I:*

- *if (g,e) is Ramsey, then $\mathtt{value}_s(g,e,I)$ holds iff $\pi(w|_I) = s$,*
- *if both $\mathtt{value}_s(g,e,I)$ and $\mathtt{value}_t(g,e,I)$ hold, then $s = t$.*

Proof. As already mentioned, we encode both functions g and e by tuples of monadic predicates. This allows us to use shorthands such as $g(x) = k$, where x is a first-order variable and $1 \leq k \leq 2|S|$, for claiming that the point x of the underlying word w is mapped via g to the number k. Similarly, we encode the convex subset I of α by a monadic predicate and we write $x \in I$ as a shorthand for a formula that states that the point x belongs to I.

We assume from now that (g,e) is Ramsey. Under this assumption, it will be clear that the constructed formulas will satisfy the desired properties. We remark, however, that the following definitions make sense also in the case when (g,e) is not Ramsey.

Given a convex subset I, we denote by $\mathtt{level}(g,I)$ the maximal value of $g(x)$ for x ranging over I. Of course, the properties $\mathtt{level}(g,I) = k$ and $\mathtt{level}(g,I) \leq k$ are definable in first-order logic.

We will construct by induction on $k \in \{0,1,\ldots,2|S|\}$ a function function $\mathtt{value}^k(g,e,I) \in S \cup \{\varepsilon\}$, and prove that it has the following properties:

- $\mathtt{value}^k(g,e,I) = s$ is first-order definable for all $s \in S \cup \{\varepsilon\}$, say by the formula $\mathtt{value}^k_s(g,e,I)$, and;
- $\mathtt{value}^k(g,e,I)$ is defined iff $\mathtt{level}(g,I) \leq k$, and in this case equals $\pi(w|I)$ (with the convention that $\pi(w|I) = \varepsilon$ iff $I = \emptyset$).

The base case is when $k = 0$, i.e., when I is empty. In this case, we set $\mathtt{value}^k(g,e,I)$ to be defined iff $I = \emptyset$ and to have value ε. Of course, this is first-order definable and satisfies the expected induction hypothesis.

Let now $k \geq 1$. We aim at constructing $\mathtt{value}^k(g,e,I)$. First, if $\mathtt{level}(g,I) < k$, then one outputs $\mathtt{value}^{k-1}(g,e,I)$. Otherwise, the convex subset I can be uniquely divided into $X < J < Y$ such that $X \cup J \cup Y = I$, and J is the minimal convex subset containing $I \cap g^{-1}(k)$. Remark that X, J, and Y are first-order definable in terms of I, g, and k. Furthermore, let f be $e(x)$ for some $x \in I \cap g^{-1}(k)$. From the assumption that I has level k for g, we know that all elements in $I \cap g^{-1}(k)$ are neighbors. This means in particular, using the Ramsey hypothesis, that $\sigma(x,y) = f$ for all $x < y$ chosen in $I \cap g^{-1}(k)$. The mapping $\texttt{value}^k(g,e,I)$ is then defined as follows:

1. if $J = \{x\}$ then
$$\texttt{value}^k(g,e,I) = \texttt{value}^{k-1}(g,e,X) \cdot w(x) \cdot \texttt{value}^{k-1}(g,e,Y) \ ,$$
2. otherwise if J has a minimal and a maximal element y, then[6]
$$\texttt{value}^k(g,e,I) = \texttt{value}^{k-1}(g,e,X) \cdot f \cdot w(y) \cdot \texttt{value}^{k-1}(g,e,Y) \ ,$$
3. if J has a minimal element but no a maximal element, then
$$\texttt{value}^k(g,e,I) = \texttt{value}^{k-1}(g,e,X) \cdot f^{\tau} \cdot \texttt{value}^{k-1}(g,e,Y) \ ,$$
4. if J has no minimal element but a maximal element y, then
$$\texttt{value}^k(g,e,I) = \texttt{value}^{k-1}(g,e,X) \cdot f^{\tau^*} \cdot w(y) \cdot \texttt{value}^{k-1}(g,e,Y) \ ,$$
5. if J has no minimal element and no maximal element, then
$$\texttt{value}^k(g,e,I) = \texttt{value}^{k-1}(g,e,X) \cdot f^{\tau^*} \cdot f^{\tau} \cdot \texttt{value}^{k-1}(g,e,Y) \ .$$

One easily checks that this function is first-order definable. It is also easy to prove that if (g,e) is Ramsey and $\texttt{level}(g,I) \leq k$, then $\texttt{value}^k(g,e,I)$ coincides with $\pi(w|I)$.

At this step, the first conclusion of the lemma is already satisfied by the formulas $\texttt{value}_s^{2|S|}(g,e,I)$. The second point, however, is false in general. Indeed, we did not pay attention so far on what the formulas compute in the case where (g,e) is not Ramsey. In particular, it can happen that both formulas $\texttt{value}_s^{2|S|}(g,e,I)$ and $\texttt{value}_t^{2|S|}(g,e,I)$ hold for distinct elements $s,t \in S$. However, this can be easily avoided using the following formula:
$$\texttt{value}_s(g,e,I) \ =^{\text{def}} \ \texttt{value}_s^{2|S|}(g,e,I) \ \wedge \ \bigwedge_{t \neq s} \neg\texttt{value}_t^{2|S|}(g,e,I) \ .$$
This formula ensures the second property of the lemma by construction, and behaves like $\texttt{value}_s$ whenever (g,e) is Ramsey. □

Lemma 5. *For all condensations $\sim$, there is X such that $\sim$ and $\approx_X$ coincide.*

Proof. It is standard that, given a linear ordering β, there exists a subset Y of β such that for all $x < y$ in β, $[x,y]$ intersects both Y and its complement $\beta \setminus Y$ (indeed, one can first prove it for scattered linear orderings and for dense linear orderings, and then combine those sub-results using the fact that every linear ordering is a dense product of non-empty scattered linear orderings [11]).

The result is then straightforward: consider Y obtained from the claim above applied to the linear ordering $\beta = \alpha/\sim$. We construct the desired set X in such

[6] Remark that this definition, as well as the next item, is not left/right symmetric. This simply reflects the asymmetry occurring in the definition of σ_w, by $\sigma(x,y) = \pi(w|[x,y))$ for all $x < y$.

a way that it contains the elements of the equivalence classes of $\sim$ that belong to Y, i.e., $X = \{x \ : \ \{y \sim x\} \in Y\}$. It is clear that $x \sim y$ iff $x \approx_X y$. $\square$

Lemma 6. *If* $\texttt{validity}(g,e)$ *holds, then* $\texttt{value}(g,e,I) = \pi(w|_I)$ *for all convex subsets* I *of* α.

Proof. Recall that, given a convex subset I of α and a condensation $\sim$ of $\alpha|_I$, $w[I,\sim]$ of domain $\beta = (\alpha|_I)/\sim$ in which every $\sim$-equivalence class J is labeled by $\texttt{value}(g,e,J)$. Suppose that $\texttt{validity}(g,e)$ holds, namely, that for all convex subsets I of α and all condensations $\sim$ of $\alpha|_I$, the following conditions are satisfied:

(C1) if I is a singleton $\{x\}$, then $\texttt{value}(g,e,I) = w(x)$,

(C2) if $w[I,\sim] = st$ for some $s,t \in S$, then $\texttt{value}(g,e,I) = s \cdot t$,

(C3) if $w[I,\sim] = s^\omega$ for some $s \in S$, then $\texttt{value}(g,e,I) = s^\tau$,

(C4) if $w[I,\sim] = s^{\omega^*}$ for some $s \in S$, then $\texttt{value}(g,e,I) = s^{\tau^*}$,

(C5) if $w[I,\sim] = P^\eta$ for some $P \subseteq S$, then $\texttt{value}(g,e,I) = P^\kappa$.

To show that $\texttt{value}(g,e,I) = \pi(w|_I)$ for all convex subsets I, we use evaluation trees. Precisely, we fix a convex subset I of α and an evaluation tree $\mathcal{T} = \langle T,\gamma\rangle$ over the word $w|_I$ (the evaluation tree exists thanks to Proposition 1), and we prove, by exploiting an induction on $\mathcal{T}$, that

$$\texttt{value}(g,e,I) \ = \ \gamma(I) \ .$$

Since $\gamma(I) = \pi(w|_I)$ (by Proposition 2), it follows that $\texttt{value}(g,e,I) = \pi(w|_I)$.

If $\mathcal{T}$ consists of a single leaf, then I is a singleton of the form $\{x\}$. Condition C1 then implies $\texttt{value}(g,e,I) = w(x) = \gamma(I)$.

If the root of $\mathcal{T}$ is not a leaf, then we let $\sim$ be the condensation of $\alpha|_I$ induced by the children of the root of $\mathcal{T}$, $\beta = (\alpha|_I)/\sim$ be the corresponding condensed order (formally, $\beta = \texttt{children}(I)$), and, for each class $J \in \beta$, we let $\mathcal{T}_J$ be the corresponding subtree of $\mathcal{T}$ (formally, $\mathcal{T}_J = \mathcal{T}|_J$). Using the induction hypothesis on each evaluation tree $\mathcal{T}_J$, we claim that $\texttt{value}(g,e,J) = \gamma(J)$ for all $J \in \beta$. Moreover, we know from the definition of $w[I,\sim]$ that $w[I,\sim](J) = \texttt{value}(g,e,J)$ for all $J \in \beta$, and hence $w[I,\sim]$ is isomorphic to the word $\prod_{J\in\beta}\gamma(J)$. We know from the definition of $\mathcal{T}$ that the image under π_0 of the word $\prod_{J\in\beta}\gamma(J)$ is defined. From this we derive that $\prod_{J\in\beta}\gamma(J)$ is isomorphic to one of the following words:

1. a finite word $s_1 ... s_n$, with $n \geq 1$ and $s_1, ..., s_n \in S$,
2. an ω-word se^ω, with $s,e \in S$ and e idempotent,
3. an ω^*-word $e^{\omega^*}s$, with $s,e \in S$ and e idempotent,
4. a dense word $sP^\eta t$, with $s,t \in S \cup \{\varepsilon\}$ and $P \subseteq S$.

We only analyze the first two cases (the arguments for the remaining cases are similar).

If $\prod_{J\in\beta}\gamma(J)$ is a finite word of the form $s_1 ... s_n$, then we let $J_1 < ... < J_n$ be the positions in it (recall that these are $\sim$-equivalence classes for $\alpha|_I$) and we observe that $\texttt{value}(g,e,J_i) = w[I,\sim](J_i) = \gamma(J_i) = s_i$ for all $1 \leq i \leq n$. We first prove, by exploiting an induction on i, that for every $1 \leq i \leq n$,

$$\texttt{value}(g,e,J_1 \cup ... \cup J_i) \;=\; s_1 \cdot ... \cdot s_i$$

The base case $i = 1$ is trivial since we already know from previous arguments that $\texttt{value}(g,e,J_1) = \gamma(J_1)$. As for the induction step, we assume that $\texttt{value}(g,e,J_1 \cup ... \cup J_i) = s_1 \cdot ... \cdot s_i$ and we prove the analogous equality for $i+1$. For this, we consider the condensation $\approx_i$ that partitions $\alpha|_{J_1 \cup ... \cup J_{i+1}}$ into the two consecutive classes $J_1 \cup ... \cup J_i$ and J_{i+1}. We have that $w[I,\approx_i] = st$, where $s = s_1 \cdot ... \cdot s_i$ and $t = s_{i+1}$. Using Condition C2, we then derive

$$\texttt{value}(g,e,I) \;=\; s \cdot t \;=\; (s_1 \cdot ... \cdot s_i) \cdot s_{i+1} \;=\; s_1 \cdot ... \cdot s_{i+1}.$$

Finally, for $i = n$, the above property implies

$$\texttt{value}(g,e,I) = \texttt{value}(g,e,J_1 \cup ... \cup J_n) = s_1 \cdot ... \cdot s_n = \pi\Big(\prod\nolimits_{J \in \beta} \gamma(J)\Big) = \gamma(I).$$

Let us now consider the case where $\prod_{J \in \beta} \gamma(J)$ is an ω-word of the form se^ω. We denote by $J_1 < J_2 < ...$ the positions in $\prod_{J \in \beta} \gamma(J)$ (recall that these are $\sim$-equivalence classes for $\alpha|_I$). We observe that $\prod_{i \geq 2} \gamma(J_i) = e^\omega$ and hence, by Condition C3, we derive

$$\texttt{value}(g,e,J_2 \cup J_3 \cup ...) \;=\; e^\tau.$$

We also know that $\texttt{value}(g,e,J_1) = s$. Let us now consider the condensation $\approx$ that partitions $\alpha|_I$ into the two consecutive classes J_1 and $(J_2 \cup J_3 \cup ...)$. We have that $w[I,\approx]$ is a word of the form st, where $t = e^\tau$, and hence, by Condition 2, we derive

$$\texttt{value}(g,e,I) \;=\; s \cdot t \;=\; s \cdot e^\tau \;=\; \pi\Big(\prod\nolimits_{J \in \beta} \gamma(J)\Big) \;=\; \gamma(I).$$

$\square$