Cano Gómez, A.; Cantero Delgado, J.; Martínez-Pastor, A. (2021). A positive extension of Eilenberg's variety theorem for non-regular languages. Applicable Algebra in Engineering Communication and Computing. 32(5):553-573. https://doi.org/10.1007/s00200-020-00414-2

Noname manuscript No.
(will be inserted by the editor)

A positive extension of Eilenberg's variety theorem for non-regular languages

A. Cano · J. Cantero · A. Martínez-Pastor

Received: date / Accepted: date

Abstract In this paper we go further with the study initiated by Behle, Krebs and Reifferscheid in 2011, who gave an Eilenberg-type theorem for non-regular languages via typed monoids. We provide a new extension of that result, inspired by the one carried out by Pin in the regular case in 1995, who considered classes of languages not necessarily closed under complement. We introduce the so-called positively typed monoids, and give a correspondence between varieties of such algebraic structures and positive varieties of possibly non-regular languages. We also prove a similar result for classes of languages with weaker closure properties.

Keywords monoids · varieties · formal languages

Mathematics Subject Classification (2010) 68Q70 · 68Q45 · 20M07 · 20M35

1 Introduction

Within the algebraic study of formal languages, as Pin [14] pointed out, the most important tool for classifying regular languages is Eilenberg's variety theorem [9], which provides a one-to-one correspondence between varieties of finite monoids and varieties of languages. Since the establishment of this celebrated theorem, many generalisations have been obtained from different points of view. For instance, in some cases languages of words are replaced by languages of different structures, such as trees [19,4,5]. Another direction is to

The third author is supported by Proyecto PGC2018-096872-B-100-AR, Agencia Estatal de Investigación (Spain), and by Proyecto Prometeo/2017/057, Generalitat Valenciana (Spain).

A. Cano, E-mail: antonio.cano.gomez@outlook.com
J. Cantero, E-mail: jecandel@hotmail.com
A. Martínez-Pastor, ORCID ID: 0000-0002-0208-4098, E-mail: anamarti@mat.upv.es
Instituto Universitario de Matemática Pura y Aplicada (IUMPA-UPV), Universitat Politècnica de València, Camino de Vera, s/n, 46022, Valencia, Spain. Tel: +34 963879662

consider different closure properties for the classes of languages and monoids [1,14,16]. A further development in [2] deals with varieties of non-regular languages and new algebraic structures. More recently, a categorical approach has been used to obtain Eilenberg-like theorems in [22] (for regular languages), and also in [18].

The starting point of our investigations is the work of Pin [14], where he extends the variety theorem to positive varieties of regular languages. A positive variety is obtained by relaxing the definition of a variety in the sense that only positive Boolean operations (union and intersection) are allowed, that is, the class of languages is not necessarily closed under complement. On the algebraic counterpart, he defines varieties of ordered monoids, which are monoids equipped with a stable order relation (see also [7,8]). Further research in the same line can be found in [16], where Polák extends Eilenberg's variety theorem to conjunctive and disjunctive varieties of languages, which are classes of languages closed under intersection and union, respectively. On the algebraic side, monoids and ordered monoids are replaced by commutative semirings. Note that the extensions made by Pin and Polák work for regular languages only.

Later on, Behle, Krebs and Reifferscheid [2] introduce a new algebraic structure, called typed monoid, to describe formal languages. This structure adds additional information to a monoid, which leads to a finer notion of language recognition. In the classical approach, a language $L \subseteq \Sigma^*$ is recognized by a monoid M if there exist a morphism $h : \Sigma^* \to M$ and a subset P of M such that $L = h^{-1}(P)$. The notion of syntatic monoid as a minimal recognizer is key in the study of regular languages. However, the syntactic monoid does not capture enough information, in particular when one goes beyond regular languages. In the new strategy developed by these authors, on the one hand it is possible to control the morphisms allowed (especially over the images of single letters) and, on the other hand, the allowed accepting sets are also limited. The union of these two ideas leads to the concept of typed monoid, which can be applied to infinite monoids and recognition of non-regular languages.

The strategy of limiting the allowed accepting subset was already used by Sakarovitch [17] for an algebraic study of context-free languages, through the category of pointed monoids. In the new approach proposed in [2] instead of adding a "point" to the monoid, a finite Boolean algebra is attached to the monoid and the accepting subsets are required to be elements of this Boolean algebra. In fact, a preliminary version of the notion of typed monoid, named finitely typed monoid, was first introduced in [11], just controlling the accepting sets but not the morphisms. In the latter case the aim was to study some circuit classes, in the framework of the known connections between formal languages, logic and circuit complexity. We refer readers interested in those connections to the book [20].

The idea of having some control on the morphisms arising in the language recognition by monoids was also considered by Pin and Straubing in [15], based on previous work of the latter author [21]. They introduced the notion of C-variety, where C is a category of admissible morphisms between finitely generated free monoids, and the elements of such varieties are not semigroups or monoids, but morphisms from finitely generated free monoids onto finite monoids, called stamps. The proposal in [2] to limit the morphisms is through the use of the so-called *units* in the definition of a typed monoid. It is to be said that one of the motivations for both developements is that some classes of languages defined through logic or circuit complexity do not form varieties in the usual sense, since they are not closed under arbitrary inverse morphisms.

The purpose of this paper is to achieve a new extension of the Eilenberg's theorem for varieties of non-regular languages by combining the above mentioned approaches. We consider classes of languages not closed under complements, as proposed by Pin, and a new algebraic object, called positively typed monoid, inspired by the one defined by Behle, Krebs and Reifferscheid. This requires the redefinition of concepts such as positively typed morphisms, recognition of languages and so on, and some adjustments in the proofs, which are mainly inspired by the ones in [2].

Also an Eilenberg-like theorem is given for classes of languages with weaker closure properties than positive varieties, such as the closure under inverse length preserving morphisms instead of arbitrary morphisms (see section 5). The role of units in the positively typed monoids is significant in the algebraic study of such classes, since it allows to give a notion of length preserving morphisms through a requirement that units are mapped to units.

Our hope is that our approach could be the appropriate tool to characterize algebraically some interesting classes of languages defined by logic which are not closed under either complements or inverse morphisms. Examples of these classes are regular classes of languages described by fragments of first order logic with modular predicates as considered in [6], or some classes of non-regular languages recognized by Parikh automata (see [10] and [3]). We will not delve here into the detailed description of such classes, but mention them as a motivation for our work.

The structure of the paper is the following: After giving some preliminaries in section 2, in section 3 we present the fundamental notions and basic results regarding positively typed monoids. In section 4 we connect these concepts with the ones dealing with recognizability and define the syntactic positively typed monoid. In section 5 we introduce positive weakly closed classes of languages and we show that they correspond to weakly closed classes of positively typed monoids. Next we present in section 6 the announced extension of the Eilenberg's variety theorem by considering positive varieties of languages and varieties of positively typed monoids. Finally, in the last section we present an example and discuss some possible further work.

2 Preliminaries

We assume the reader to be familiar with the basic notions of monoids, language recognition and varieties, and, more specifically, with Eilenberg's variety theory. Our main reference on this topic is the book [13] and our notation will follow mainly this source (see also [12]). For the sake of completeness, we review some notions.

An *alphabet* Σ is a finite nonempty set whose elements are called *letters*. The elements of the *free monoid* Σ^* are called *words* and a *language* is any subset $L \subseteq \Sigma^*$, that is, any set of words over Σ. The length of a word u is denoted by $|u|$. If a is a letter, $|u|_a$ denotes the number of occurrences of a in u. Given two alphabets, Σ and Δ, a morphism $\varphi : \Sigma^* \longrightarrow \Delta^*$ is called *length preserving* if $\varphi(\Sigma) \subseteq \Delta$, i.e., if it maps letters to letters. If $L \subseteq \Sigma^*$ is a language and M is a monoid, M *recognizes* L if there exist a subset P of M and a monoid morphism $\varphi : \Sigma^* \longrightarrow M$ such that $L = \varphi^{-1}(P)$.

The *syntactic congruence* of a language $L \subseteq \Sigma^*$ is defined as: for words x and y, $x \sim_L y$ iff for every $u, v \in \Sigma^*$ it holds $uxv \in L \Leftrightarrow uyv \in L$. The corresponding quotient monoid of Σ^* is called the *syntactic monoid* of L, denoted by $M(L)$. The canonical epimorphism associated to this congruence is called *syntactic morphism* and denoted by ρ_L.

We introduce some basic notions we need to define our main algebraic structure, the positively typed monoids.

Consider a nonempty set S. A *positive Boolean algebra* $\mathcal{A}$ over S is a set of subsets of S which is closed under finite union and finite intersection. Note that a positive Boolean algebra always contains the empty set and the full set since $\emptyset = \cup_{i\in\emptyset} S_i$ and $S = \cap_{i\in\emptyset} S_i$. The intersection of all positive Boolean algebras containing a set $\mathcal{F}$ of subsets of S is again a positive Boolean algebra, called the *positive Boolean algebra generated by* $\mathcal{F}$.

Let $\mathcal{A}, \mathcal{B}$ be positive Boolean algebras over S and T, respectively. A morphism h from $\mathcal{A}$ on $\mathcal{B}$ is an application $h : \mathcal{A} \longrightarrow \mathcal{B}$, such that: $h(\emptyset) = \emptyset; h(S) = T; h(A_1 \cup A_2) = h(A_1) \cup h(A_2); h(A_1 \cap A_2) = h(A_1) \cap h(A_2)$, for every $A_1, A_2 \in \mathcal{A}$.

A *Boolean algebra* over S is a positive Boolean algebra which is also closed under complements.

As usual, we will denote by $\mathbb{Z}$ the infinite monoid consisting of the integers with addition, which is finitely generated by the set $\{1, -1\}$. We also denote by $\mathbb{Z}^+$ the set of positive integers and by $\mathbb{Z}_0^+$ the set $\mathbb{Z}^+ \cup \{0\}$. For each positive integer m, we denote by $\mathbb{Z}_m$ the corresponding monoid of integers modulo m.

3 Positively typed monoids

We establish in this section the concept of positively typed monoid and the corresponding morphisms. Following the approach in [2], and starting from the canonical definition of language recognition, our first aim will be to limit the allowed accepting sets. To do this, we require them to be elements of a finite positive Boolean algebra over the initial monoid. Moreover, we wish to control the morphisms allowed, specially to rule where the single letters are mapped to, which will be helpful when dealing with infinite monoids or non-regular languages. So, we also add to our structure a finite subset of the considered monoid and require morphisms to map elements of this subset on themselves.

Definition 1 A *positively typed monoid* is a triple $(S, \mathcal{A}, \mathcal{E})$, where S is a finitely generated monoid, $\mathcal{A}$ is a finite positive Boolean algebra over S, and $\mathcal{E}$ is a finite subset of S. The elements of $\mathcal{A}$ are called *(positive) types* and the elements of $\mathcal{E}$ are called the *units*.

We say that a positively typed monoid $(S, \mathcal{A}, \mathcal{E})$ is *free*, if the monoid S is a free monoid.

If the positive Boolean algebra $\mathcal{A}$ is generated by a single set S_1, then we will write $(S, S_1, \mathcal{E})$ instead of $(S, \{S, S_1, \emptyset\}, \mathcal{E})$.

Note that the above definition extends the one of typed monoid $(S, \mathcal{A}, \mathcal{E})$ in [2], where $\mathcal{A}$ is a Boolean algebra instead of a positive Boolean algebra, i.e., $\mathcal{A}$ is closed under unions, intersections and complements. Hence all examples of typed monoids are, in particular, examples of positively typed monoids.

Example 1 Given a language $L \subseteq \Sigma^*$, we can build the positively typed monoid $(\Sigma^*, L, \Sigma) = (\Sigma^*, \{\Sigma^*, L, \emptyset\}, \Sigma)$. Note that, in our case, in contrast to what happens for typed monoids, a language L and its complement L^c do not lead to the same positively typed monoid.

Observe also that in this example the units are precisely the letters in the corresponding alphabet, so every element has a unique representation as a product of the units. However, in our general definition of a positively typed monoid $(S, \mathcal{A}, \mathcal{E})$ we do not require the units $\mathcal{E}$ to be generators of the monoid S.

Now we introduce the notion of morphism between positively typed monoids. The lack of complements in the positive Boolean algebra causes that our definition needs to be amended with respect to the one given for typed monoids.

Definition 2 A *positively typed morphism* $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (S', \mathcal{A}', \mathcal{E}')$ of positively typed monoids is a triple (h_m, h_a, h_u), where $h_m : S \longrightarrow S'$ is a morphism of monoids, $h_a : \mathcal{A} \longrightarrow \mathcal{A}'$ is a morphism of positive Boolean algebras, and $h_u : \mathcal{E} \longrightarrow \mathcal{E}'$ is a mapping of sets, such that the following conditions hold:

1) $h_m^{-1}(h_a(S_i)) = S_i$, $\forall S_i \in \mathcal{A}$,
2) $h_m(w) = h_u(w)$, $\forall w \in \mathcal{E}$.

Note that we use subscripts m, a and u to denote the monoid morphism, the morphism of positive Boolean algebras, and the mapping for units, respectively. The above definition guarantees that all these mappings are compatible. The following Lemma analyzes the scope of this definition.

Lemma 1 *Let $(S, \mathcal{A}, \mathcal{E})$ and $(S', \mathcal{A}', \mathcal{E}')$ be positively typed monoids. Let $h_m : S \longrightarrow S'$ and $h_a : \mathcal{A} \longrightarrow \mathcal{A}'$ be a morphism of monoids and a morphism of positive Boolean algebras, respectively, satisfying the following condition:*

1) $h_m^{-1}(h_a(S_i)) = S_i$, $\forall S_i \in \mathcal{A}$.

Then the following conditions hold:

2) $h_m(S_i) = h_a(S_i) \cap h_m(S)$, $\forall S_i \in \mathcal{A}$.
3) $h_m(S_i) \subseteq h_a(S_i)$, $\forall S_i \in \mathcal{A}$.

Assume, in addition, that $\mathcal{A}$ is a Boolean algebra, i.e. $\mathcal{A}$ is also closed under taking complements (this is to say that $(S, \mathcal{A}, \mathcal{E})$ is a typed monoid). Then all conditions 1) - 3) are equivalent.

Proof We first prove that 1) $\Rightarrow$ 2). Assume that 1) holds and let $S_i \in \mathcal{A}$. Then $h_m^{-1}(h_a(S_i) \cap h_m(S)) = h_m^{-1}(h_a(S_i)) \cap h_m^{-1}(h_m(S)) = h_m^{-1}(h_a(S_i)) \cap S = S_i$. Since h_m is surjective when co-restricted to $h_m(S)$ it holds $h_m(S_i) = h_m(h_m^{-1}(h_a(S_i) \cap h_m(S))) = h_a(S_i) \cap h_m(S)$, as desired.

It is clear that 2) $\Rightarrow$ 3).

Now assume that $\mathcal{A}$ is also closed under taking complements.

3) $\Rightarrow$ 2). Assume that 3) holds but there exists some $S_i \in \mathcal{A}$ such that $h_m(S_i) \neq h_a(S_i) \cap h_m(S)$. Since $h_m(S_i) \subseteq h_a(S_i) \cap h_m(S)$, this means that there exists $x \in h_a(S_i) \cap h_m(S)$ such that $x \notin h_m(S_i)$. But $x \in h_m(S)$ implies that $x = h_m(y)$ with $y \in S$. Since $x \notin h_m(S_i)$, then $y \notin S_i$ and so $x \in h_m(S_i^c)$. But $\mathcal{A}$ is closed under taking complements, so S_i^c is again a type and, therefore, $x \in h_m(S_i^c) \subseteq h_a(S_i^c)$. Since $x \in h_a(S_i)$ too, this implies that $x \in h_a(S_i) \cap h_a(S_i^c) = h_a(S_i \cap S_i^c) = h_a(\emptyset) = \emptyset$. This is a contradiction so the claim follows.

3) $\Rightarrow$ 1). Assume 3) holds. We claim first that under this assumption, it holds that $h_m^{-1}(h_m(S_i)) = S_i$, $\forall S_i \in \mathcal{A}$. If this is not true, then there exists some $S_i \in \mathcal{A}$ such that $S_i \subsetneq h_m^{-1}(h_m(S_i))$. Hence there exists $x \in h_m^{-1}(h_m(S_i))$, such that $x \notin S_i$, so $x \in S_i^c$. Now, $x \in h_m^{-1}(h_m(S_i))$ implies that $h_m(x) \in h_m(S_i) \subseteq h_a(S_i)$. On the other hand, $x \in S_i^c$ implies that $h_m(x) \in h_m(S_i^c)$. But S_i^c is again a type since $\mathcal{A}$ is a Boolean algebra, so it holds $h_m(S_i^c) \subseteq h_a(S_i^c)$ by 3). Hence it follows that $h_m(x) \in h_a(S_i) \cap h_a(S_i^c) = h_a(S_i \cap S_i^c) = \emptyset$, which is a contradiction. So the claim follows.

Now let any $S_i \in \mathcal{A}$. We have just proved that $h_m^{-1}(h_m(S_i)) = S_i$. But since 3) is equivalent to 2), we also have that $h_m(S_i) = h_a(S_i) \cap h_m(S)$. Hence $S_i = h_m^{-1}(h_m(S_i)) = h_m^{-1}(h_a(S_i) \cap h_m(S)) = h_m^{-1}(h_a(S_i))$ and we are done.

Note that the definition of typed morphism of typed monoids which appears in [2] requires that Condition 2) in the above Lemma holds. We have replaced this in our definition of positively typed morphism by Condition 1). Hence, the above result shows, in particular, that our definition extends the one of typed morphism in [2]. Indeed we see in the next example that the equivalence of all conditions 1) - 3) fails to be true under the weaker hypotheses we are considering.

Example 2 We see next that the condition that $\mathcal{A}$ is a Boolean algebra is necessary to prove that 3) $\Rightarrow$ 1) in the above Lemma. Consider the positively typed monoids $(\mathbb{Z}_4, \{0\}, \{0\})$, $(\mathbb{Z}_4, \{0, 1\}, \{0\})$ and the mapping $h : (\mathbb{Z}_4, \{0\}, \{0\}) \longrightarrow (\mathbb{Z}_4, \{0, 1\}, \{0\})$ given in the following way:

- $h_m : \mathbb{Z}_4 \longrightarrow \mathbb{Z}_4$ is defined by $h_m(x) = 2x \pmod 4$,

- $h_a : \{\emptyset, \{0\}, \mathbb{Z}_4\} \longrightarrow \{\emptyset, \{0,1\}, \mathbb{Z}_4\}$ is given by $h_a(\emptyset) = \emptyset$, $h_a(\{0\}) = \{0,1\}$ and $h_a(\mathbb{Z}_4) = \mathbb{Z}_4$,
- $h_u(0) = 0$.

Then $h_m(\{0\}) = h_a(\{0\}) \cap h_m(\mathbb{Z}_4) = \{0,1\} \cap \{0,2\} = \{0\}$, but $\{0\} \neq h_m^{-1}(h_a(\{0\})) = \{0,2\}$. Therefore $h_m(S_i) = h_a(S_i) \cap h_m(S)$ does not imply that $h_m^{-1}(h_a(S_i)) = S_i$ for every type S_i in $\mathcal{A}$.

Remark 1 Let $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (S', \mathcal{A}', \mathcal{E}')$ be a morphism of positively typed monoids. Then note that the following facts hold:

a) The morphism of monoids h_m maps units to units and is compatible with the mapping h_u; in particular, h_m induces h_u.
b) The morphism h_m also respects the union and the intersection of types, because of Condition 2) in the previous Lemma and the fact that h_a does so.
c) If the morphism of monoids $h_m : S \longrightarrow S'$ is surjective and Condition 2) holds, then $h_m(S_i) = h_a(S_i) \cap h_m(S) = h_a(S_i)$, and so $h_m^{-1}(h_a(S_i)) = h_m^{-1}(h_m(S_i)) = S_i$, i.e. Condition 1) holds. Hence in this case all conditions 1) - 3) in Lemma 1 are equivalent and, indeed, they are equivalent to: $h_m(S_i) = h_a(S_i)$, $\forall S_i \in \mathcal{A}$. In particular, if h_m is surjective, then h_m also induces h_a.
d) The morphism of positive Boolean algebras h_a is injective, since if $h_a(S_i) = h_a(S_j)$ with $S_i, S_j \in \mathcal{A}$, then $h_m^{-1}(h_a(S_i)) = h_m^{-1}(h_a(S_j))$, which implies that $S_i = S_j$. This means, in particular, that $|\mathcal{A}| \leq |\mathcal{A}'|$.

Definition 3 Let $(S, \mathcal{A}, \mathcal{E}), (T, \mathcal{I}, \mathcal{F})$ be positively typed monoids.

– A positively typed morphism $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ is *injective* (resp. surjective, bijective) if h_m, h_a, h_u are all injective (resp. surjective, bijective).
– A positively typed monoid $(T, \mathcal{I}, \mathcal{F})$ is a *positively typed submonoid* of $(S, \mathcal{A}, \mathcal{E})$ if T is a submonoid of S and there exists an injective positively typed morphism from $(T, \mathcal{I}, \mathcal{F})$ to $(S, \mathcal{A}, \mathcal{E})$.
– $(T, \mathcal{I}, \mathcal{F})$ *divides* $(S, \mathcal{A}, \mathcal{E})$ if $(T, \mathcal{I}, \mathcal{F})$ is the epimorphic image of a submonoid of $(S, \mathcal{A}, \mathcal{E})$. In this case, we will write $(T, \mathcal{I}, \mathcal{F}) \preceq (S, \mathcal{A}, \mathcal{E})$.

Remark 2 As expected, given a positively typed morphism $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$, the image of h:

$$h((S, \mathcal{A}, \mathcal{E})) = (h_m(S), \{h_a(S_1) \cap h_m(S), S_i \in \mathcal{A}\}, h_u(\mathcal{E})),$$

is a submonoid of $(T, \mathcal{I}, \mathcal{F})$. Analogously, the preimage of h:

$$h^{-1}((T, \mathcal{I}, \mathcal{F})) = (h_m^{-1}(T), \{h_a^{-1}(T_i), T_i \in \mathcal{I}\}, h_u^{-1}(\mathcal{F})),$$

is a submonoid of $(S, \mathcal{A}, \mathcal{E})$ (recall that h_a is injective).

Lemma 2 *Let $(S, \mathcal{A}, \mathcal{E}), (T, \mathcal{I}, \mathcal{F}), (U, \mathcal{P}, \mathcal{G})$ be positively typed monoids and let $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$, and $f : (T, \mathcal{I}, \mathcal{F}) \longrightarrow (U, \mathcal{P}, \mathcal{G})$ be positively typed morphisms. Then the composition $f \circ h$ is a positively typed morphism from $(S, \mathcal{A}, \mathcal{E})$ to $(U, \mathcal{P}, \mathcal{G})$.*

Proof We prove only Condition 1) for $f \circ h$, since Condition 2) is trivial. Let $S_i \in \mathcal{A}$. Since h is a positively typed morphism, then $h_m^{-1}(h_a(S_i)) = S_i$. Moreover $h_a(S_i)$ is a type of $(T, \mathcal{I}, \mathcal{F})$ and f is a positively typed morphism too, hence $f_m^{-1}(f_a(h_a(S_i))) = h_a(S_i)$, and then $h_m^{-1}(f_m^{-1}(f_a(h_a(S_i))) = h_m^{-1}(h_a(S_i)) = S_i$.

Example 3 Consider the positively typed monoids:

- $(\Sigma^*, L_{even}, \{a\})$, where L_{even} is the set of all words of even length over the alphabet $\Sigma = \{a\}$,
- $(S, \mathcal{A}, \mathcal{E}) = (\mathbb{Z}_4, \{\emptyset, \{0\}, \mathbb{Z}_4\}, \{1\})$,
- $(T, \mathcal{I}, \mathcal{F})$= $(\mathbb{Z}_2, \{\emptyset, \{0\}, \mathbb{Z}_2\}, \{1\})$,
- $(U, \mathcal{P}, \mathcal{G}) = (\mathbb{Z}_4, \{\emptyset, \{0, 2\}, \mathbb{Z}_4\}, \{1\})$.

We can define a surjective positively typed morphism h from $(\Sigma^*, L_{even}, \{a\})$ to $(T, \mathcal{I}, \mathcal{F})$, where $h_m(w) = |w| \pmod 2$, for any $w \in \Sigma^*$, h_a acts in such a way that $h_a(L_{even}) = \{0\}$ and $h_u(a) = 1$. It is straightforward to check that Conditions 1) and 2) of the definition hold.

On the other hand, there exists also a surjective positively typed morphism g from $(\Sigma^*, L_{even}, \{a\})$ to $(U, \mathcal{P}, \mathcal{G})$, where: $g_m(w) = |w| \pmod 4$, for any $w \in \Sigma^*$, g_a acts so that $g_a(L_{even}) = \{0, 2\}$, and $g_u(a) = 1$. Conditions 1) and 2) of the definition are clearly satisfied.

However, there is no positively typed morphism h from $(\Sigma^*, L_{even}, \{a\})$ to $(S, \mathcal{A}, \mathcal{E})$. If such a morphism h would exist, then it would satisfy $\emptyset \neq h_m(L_{even}) \subseteq h_a(L_{even}) = \{0\}$ and $h_m(a) = h_u(a) = 1$. Then, either $h_m(w) = 1$, or $h_m(w) = |w| \pmod 4$, for any word $w \in \Sigma^* \setminus L_{even}$. Clearly, none of these choices provides a monoid morphism h_m from Σ^* to $\mathbb{Z}_4$.

We see now that there is no positively typed morphism from $(S, \mathcal{A}, \mathcal{E})$ to $(T, \mathcal{I}, \mathcal{F})$. If such a morphism h would exist, then clearly $h_m(0) = 0$ and $h_m(1) = h_u(1) = 1$. The monoid morphism $h_m(x) = x \pmod 2$ does not define a positively typed monoid, since it holds $h_m^{-1}(h_a(\{0\})) = \{0, 2\} \neq \{0\}$. In fact, any monoid morphism such that $h_m(x) = 0$ for some $x \neq 0$ does not define a positively typed morphism, since Condition 1) fails. But the other possible mappings from $\mathbb{Z}_4$ to $\mathbb{Z}_2$ do not give a monoid morphism. Similarly, it is easy to see that there is no positively typed morphism from $(T, \mathcal{I}, \mathcal{F})$ to $(S, \mathcal{A}, \mathcal{E})$.

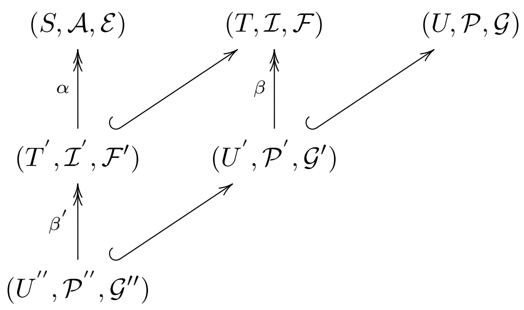
Finally, note that the mapping $f : (U, \mathcal{P}, \mathcal{G}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ given by $f_m(x) = x \pmod 2$, for every $x \in \mathbb{Z}_4$, $f_a(\{0, 2\}) = \{0\}$ and $f_u(1) = 1$, defines a surjective positively typed monoid. In particular $(T, \mathcal{I}, \mathcal{F})$ divides $(U, \mathcal{P}, \mathcal{G})$.

Lemma 3 *If $(S, \mathcal{A}, \mathcal{E})$ divides $(T, \mathcal{I}, \mathcal{F})$ and $(T, \mathcal{I}, \mathcal{F})$ divides $(U, \mathcal{P}, \mathcal{G})$, then $(S, \mathcal{A}, \mathcal{E})$ divides $(U, \mathcal{P}, \mathcal{G})$.*

Proof By definition, there exist positively typed submonoids $(T^{'}, \mathcal{I}^{'}, \mathcal{F}^{'})$ of $(T, \mathcal{I}, \mathcal{F})$, and $(U^{'}, \mathcal{P}^{'}, \mathcal{G}')$ of $(U, \mathcal{P}, \mathcal{G})$, respectively, and surjective morphisms $\alpha : (T^{'}, \mathcal{I}^{'}, \mathcal{F}^{'}) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ and $\beta : (U^{'}, \mathcal{P}^{'}, \mathcal{G}') \longrightarrow (T, \mathcal{I}, \mathcal{F})$.

We denote $(U^{''}, \mathcal{P}^{''}, \mathcal{G}^{''}) = \beta^{-1}((T^{'}, \mathcal{I}^{'}, \mathcal{F}^{'}))$, and $\beta^{'}$ is the restriction of β to $(U^{''}, \mathcal{P}^{''}, \mathcal{G}^{''})$. Then $\alpha \circ \beta^{'}$ is a surjective positively typed morphism from

$(U^{''},\mathcal{P}^{''},\mathcal{G}^{''})$ onto $(S,\mathcal{A},\mathcal{E})$. Since $(U^{''},\mathcal{P}^{''},\mathcal{G}^{''})$ is a positively typed submonoid of $(U,\mathcal{P},\mathcal{G})$, then $(S,\mathcal{A},\mathcal{E})$ divides $(U,\mathcal{P},\mathcal{G})$.



The previous Lemma says that division is a preorder in the class of positively typed monoids. However, it is not in general a partial order, as in the finite monoids case. It is not hard to construct examples of two (positively) typed monoids, say $(S,\mathcal{A},\mathcal{E})$ and $(T,\mathcal{I},\mathcal{F})$, such that $(S,\mathcal{A},\mathcal{E})$ divides $(T,\mathcal{I},\mathcal{F})$, $(T,\mathcal{I},\mathcal{F})$ divides $(S,\mathcal{A},\mathcal{E})$ but they are not isomorphic (see [2, Example 4.4]).

Next we define the notion of congruences for our new algebraic object, which are closely connected to morphisms. Beside the compatibility with the monoid's operation, our congruences must be compatible with the set of types.

Definition 4 Let $(S,\mathcal{A},\mathcal{E})$ be a positively typed monoid. A congruence $\sim$ over S is a *positively typed congruence* if

$$\forall S_i \in \mathcal{A}, s_1, s_2 \in S: \; s_1 \sim s_2 \wedge s_1 \in S_i \Rightarrow s_2 \in S_i.$$

We say in this case that $\sim$ is finer than $\mathcal{A}$.

Let $(S,\mathcal{A},\mathcal{E})$ be a positively typed monoid, and let $\sim$ be a positively typed congruence on it. Consider:

$$S_i/_\sim = \{[x]_\sim \mid x \in S_i\}; \quad \mathcal{A}/_\sim = \{S_i/_\sim \;\mid\; S_i \in \mathcal{A}\};$$

$$\mathcal{E}/_\sim = \{[x]_\sim \mid x \in \mathcal{E}\}.$$

Since $\sim$ is finer than $\mathcal{A}$, then $(S,\mathcal{A},\mathcal{E})/_\sim := (S/_\sim, \mathcal{A}/_\sim, \mathcal{E}/_\sim)$ is a positively typed monoid, called the *positively typed quotient monoid* of $(S,\mathcal{A},\mathcal{E})$ by $\sim$.

The canonical projection mapping from $(S,\mathcal{A},\mathcal{E})$ onto $(S,\mathcal{A},\mathcal{E})/\sim$ is clearly a positively typed morphism. Conversely, any positively typed morphism h on a positively typed monoid $(S,\mathcal{A},\mathcal{E})$ defines a positively typed congruence via $s_1 \sim_h s_2 \Leftrightarrow h(s_1) = h(s_2)$. This congruence is called the *kernel of h.*

Example 4 Let $(S,\mathcal{A},\mathcal{E})$ be a positively typed monoid. We can define, in particular, a congruence $\sim_{S_i}$ associated to a fixed type S_i, as follows:

$$s_1 \sim_{S_i} s_2 \Leftrightarrow (us_1v \in S_i \Leftrightarrow us_2v \in S_i, \forall u, v \in S).$$

The congruence $\sim_{S_i}$ is called the *syntactic congruence* associated to the type S_i.

In particular, in the case when $\mathcal{A}$ is the positive algebra generated by S_i, i.e., $\mathcal{A} = \{\emptyset, S_i, S\}$, it holds that $\sim_{S_i}$ is a positively typed congruence and $(S, \mathcal{A}, \mathcal{E})/_{\sim_{S_i}}$ is the *syntactic positively typed monoid* of S_i. The canonical projection $\rho_{S_i} : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (S, \mathcal{A}, \mathcal{E})/_{\sim_{S_i}}$ is called the *syntactic positively typed morphism* associated to S_i.

As we will see in the next section, a relevant particular case will appear for the positively typed monoid (Σ^*, L, Σ) associated to a language $L \subseteq \Sigma$.

4 Recognizability and syntactic positively typed monoid

In this section we deal with the recognizability of languages via positively typed monoids. As mentioned in the introduction, the allowed monoid morphisms are limited by requiring that they map letters to units and, moreover, only types are allowed to be accepting sets.

Definition 5 A positively typed monoid $(S, \mathcal{A}, \mathcal{E})$ *recognizes* a language $L \subseteq \Sigma^*$ if there exist a monoid morphism $h : \Sigma^* \longrightarrow S$ with $h(\Sigma) \subseteq \mathcal{E}$, and a type $S_i \in \mathcal{A}$ such that $L = h^{-1}(S_i)$.

Lemma 4 *A positively typed monoid $(S, \mathcal{A}, \mathcal{E})$ recognizes a language L if and only if there exists a positively typed morphism $h : (\Sigma^*, L, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$.*

Proof Assume first that $(S, \mathcal{A}, \mathcal{E})$ recognizes a language $L \subseteq \Sigma^*$. Then there exist a monoid morphism $h_m : \Sigma^* \longrightarrow S$ with $h_m(\Sigma) \subseteq \mathcal{E}$, and a type $S_i \in \mathcal{A}$ such that $L = h_m^{-1}(S_i)$. We choose such a type S_i and we define a positively typed morphism $h : (\Sigma^*, L, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ as follows: $h_m : \Sigma^* \longrightarrow S$ is the given morphism of monoids, $h_u : \Sigma \longrightarrow \mathcal{E}$ is the restriction of h_m to Σ (which is well defined since $h_m(\Sigma) \subseteq \mathcal{E}$), and h_a is given by $h_a(\emptyset) = \emptyset$, $h_a(\Sigma^*) = S$ and $h_a(L) = S_i$, where S_i is the chosen type in A such that $L = h_m^{-1}(S_i)$. Now we show the positively typed morphism conditions for h. For the first condition: $h_m^{-1}(h_a(\emptyset)) = h_m^{-1}(\emptyset) = \emptyset$, $h_m^{-1}(h_a(\Sigma^*)) = h_m^{-1}(S) = \Sigma^*$, and $h_m^{-1}(h_a(L)) = h_m^{-1}(S_i) = L$. The condition for units is obvious.

Now let $h : (\Sigma^*, L, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ be a positively typed morphism. Let $h_m : \Sigma^* \longrightarrow S$ be the monoid morphism and let $S_i = h_a(L) \in \mathcal{A}$. Then $h_m^{-1}(S_i) = h_m^{-1}(h_a(L)) = L$, since L is a type. Moreover, $h_m(\Sigma) = h_u(\Sigma) \subseteq \mathcal{E}$. So the claim follows.

Remark 3 a) Note that this definition extends the one for typed monoids, which in turn allowed to retrieve the standard notion of language recognition for finite monoids. Observe that a typed monoid $(S, \mathcal{A}, \mathcal{E})$ which recognizes a language $L \subseteq \Sigma^*$ via a given typed morphism h, also recognizes its complement L^c via the same morphism h. This makes a main difference with our notion of recognizability via positively typed monoids (see Example 5 b) below).

b) The notion of language recognition for finite ordered monoids introduced by Pin in [14] coincides with our notion in the following way: given a finite ordered monoid $(S, \leq)$, let $\mathcal{A}$ be the set of order ideals of $(S, \leq)$; then $(S, \leq)$ and the positively typed monoid $(S, \mathcal{A}, S)$ recognize exactly the same languages.

Example 5 a) Consider the language L_{even} consisting of all words of even length over the alphabet $\Sigma = \{a\}$. The examples in the previous section show that L_{even} is recognized both by the positively typed monoids:

- $(T, \mathcal{I}, \mathcal{F})= (\mathbb{Z}_2, \{\emptyset, \{0\}, \mathbb{Z}_2\}, \{1\})$, and
- $(U, \mathcal{P}, \mathcal{G})= (\mathbb{Z}_4, \{\emptyset, \{0, 2\}, \mathbb{Z}_4\}, \{1\})$.

However, the positively typed monoid $(S, \mathcal{A}, \mathcal{E}) = (\mathbb{Z}_4, \{\emptyset, \{0\}, \mathbb{Z}_4\}, \{1\})$ does not recognize this language.

b) Consider the alphabet $\Sigma = \{a, b\}$ and the context-free language

$$L_{maj} = \{x \in \Sigma^* \mid |x|_a > |x|_b\}.$$

Define the monoid morphism $h : \Sigma^* \longrightarrow \mathbb{Z}$ by $h(a) = 1$ and $h(b) = -1$. Then $h(\Sigma) \subseteq \mathbb{Z}$ and $L_{maj} = \{x \in \Sigma^* \mid h(x) > 0\} = h^{-1}(\mathbb{Z}^+)$. Hence a positively typed monoid recognizing L_{maj} is $(\mathbb{Z}, \{\mathbb{Z}, \mathbb{Z}^+, \emptyset\}, \{1, -1\})$. Observe that this positively typed monoid can not recognize $(L_{maj})^c$.

c) Let again $\Sigma = \{a, b\}$ and consider the non-regular language

$$L_{eq} = \{x \in \Sigma^* \mid |x|_a = |x|_b\}.$$

A positively typed monoid recognizing L_{eq} is $(\mathbb{Z}, \{\mathbb{Z}, \{0\}, \emptyset\}, \{1, -1\})$, via the monoid morphism $h : \Sigma^* \longrightarrow \mathbb{Z}$ given by $h(a) = 1$ and $h(b) = -1$.

Two positively typed monoids that divide each other do not need to be isomorphic, but we see next that they recognize the same languages. We will prove this with the help of the notion of syntactic positively typed monoid, which is a particular case of the construction given in Example 4.

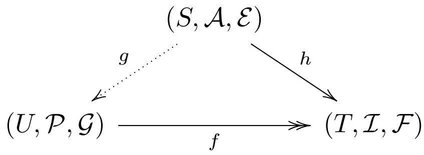
Definition 6 Let $L \subseteq \Sigma^*$ be a language and consider the syntactic congruence associated to L, i.e.:

$$x \sim_L y \Leftrightarrow (uxv \in L \Leftrightarrow uyv \in L, \ \forall u, v \in \Sigma^*).$$

Then $syn(L) = (\Sigma^*, L, \Sigma)/_{\sim_L}$ is the *syntactic positively typed monoid* of L and $\rho_L : (\Sigma^*, L, \Sigma) \longrightarrow (\Sigma^*, L, \Sigma)/_{\sim_L}$ is the *syntactic positively typed morphism.*

Clearly, $syn(L)$ recognizes L via the syntactic positively typed morphism ρ_L. As in the finite case we will show next that $syn(L)$ is the minimal positively typed monoid (with respect to division) recognizing the language L, which we will prove using some technical lemmas.

Lemma 5 *Let $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ be a positively typed morphism where $(S, \mathcal{A}, \mathcal{E})$ is a free positively typed monoid, and $\mathcal{E}$ is the generator set for S. Let $f : (U, \mathcal{P}, \mathcal{G}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ be a surjective positively typed morphism. Then* *there exists a positively typed morphism* $g : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (U, \mathcal{P}, \mathcal{G})$ *such that* $h = f \circ g$.

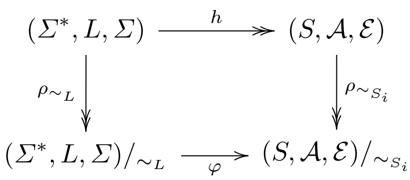


Proof Note that f_m, f_a, f_u are all surjective maps, since f is surjective. Moreover, f_a is a bijective map.

For each unit $s \in \mathcal{E}$ there exists an element $u \in \mathcal{G} \subseteq U$, such that $f_m(u) = f_u(u) = h_m(s)$, because f is surjective and $h_m(s) = h_u(s) \in \mathcal{F}$. For each $s \in \mathcal{E}$ we choose such an element, denoted by u_s. Then we define a mapping $g_u : \mathcal{E} \longrightarrow \mathcal{G}$ by setting $g_u(s) = u_s$. In particular, this defines a mapping from $\mathcal{E}$ to U which can be extended to a monoid morphism, say $g_m : S \longrightarrow U$, which verifies $h_m = f_m \circ g_m$ and also $g_m(\mathcal{E}) = g_u(\mathcal{E}) \subseteq \mathcal{G}$. Now, we define a morphism of positive Boolean algebras $g_a : \mathcal{A} \longrightarrow \mathcal{P}$ by $g_a = f_a^{-1} \circ h_a$ (recall that f_a is bijective).

We consider $g = (g_m, g_a, g_u) : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (U, \mathcal{P}, \mathcal{G})$ and we will prove that g is a positively typed morphism. Let S_i a type in $\mathcal{A}$. We have to prove that $g_m^{-1}(g_a(S_i)) = S_i$. Since $g_a(S_i)$ is a type in $(U, \mathcal{P}, \mathcal{G})$ and $f : (U, \mathcal{P}, \mathcal{G}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ is a surjective positively typed morphism, we have that $g_a(S_i) = f_m^{-1}(f_a(g_a(S_i)))$. But $f_a(g_a(S_i)) = h_a(S_i)$, so it follows that $g_a(S_i) = f_m^{-1}(h_a(S_i))$. Applying now that $f_m \circ g_m = h_m$, we deduce that $g_m^{-1}(g_a(S_i)) = g_m^{-1}(f_m^{-1}(h_a(S_i))) = h_m^{-1}(h_a(S_i)) = S_i$, where the last assertion follows from the fact that S_i is a type in $\mathcal{A}$ and $h : (S, \mathcal{A}, \mathcal{E}) \longrightarrow (T, \mathcal{I}, \mathcal{F})$ is a positively typed morphism. Hence g is a positively typed morphism and clearly $h = f \circ g$.

Lemma 6 *Let* $(S, \mathcal{A}, \mathcal{E})$ *be a positively typed monoid which recognizes a language* $L \subseteq \Sigma^*$ *by a surjective positively typed morphism* h. *Then there is an isomorphism* $\varphi : (\Sigma^*, L, \Sigma)/\sim_L \longrightarrow (S, \mathcal{A}, \mathcal{E})/\sim_{S_i}$ *with* $\varphi \circ \rho_{\sim_L} = \rho_{\sim_{S_i}} \circ h$, *where* S_i *is the type in* $\mathcal{A}$ *such that* $L = h_m^{-1}(S_i)$, *and the respective* ρ*'s are the canonical projections.*

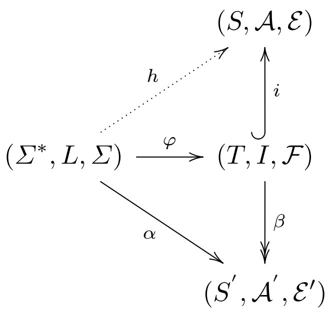


Proof Note that the fact that h is surjective implies that S_i is unique, and also that $\mathcal{A} = \{\emptyset, S_i, S\}$ and so $\sim_{S_i}$ is a positively typed congruence. Observe that for any $w \in \Sigma^*$, the condition $pwq \in L$ for $p, q \in \Sigma^*$ is equivalent to $h_m(p)h_m(w_i)h_m(q) \in S_i$. Then, for any pair $w_1, w_2 \in \Sigma^*$, it holds that $w_1 \sim_L w_2$ if and only if $h_m(w_1) \sim_{S_i} h_m(w_2)$. Then, if we define $\varphi_m : \Sigma^*/_{\sim_L} \longrightarrow$

$S/_{\sim_{S_i}}$ such that $\varphi_m([w_i]_{\sim_L}) = [h_m(w_i)]_{\sim_{S_i}}$, we have that φ_m is a monoid morphism which is well defined and injective. Clearly, $\varphi_m(\Sigma/_{\sim_L}) \subseteq \mathcal{E}/_{\sim_{S_i}}$. Moreover, since $h_m : \Sigma^* \longrightarrow S$ is surjective we deduce that φ_m is an isomorphism of monoids. Now, we define a morphism of positive Boolean algebras by the composition $\varphi_a = \rho_{a_{\sim_{S_i}}} \circ h_a \circ \rho^{-1}_{a_{\sim_L}}$. Then it holds that $\varphi = (\varphi_m, \varphi_a, \varphi_u)$ is a positive typed morphism, since $\varphi_m^{-1}(\varphi_a(L/_{\sim_L})) = \varphi_m^{-1}(h_a(L)/_{\sim_{S_i}}) = \varphi_m^{-1}(S_i/_{\sim_{S_i}})$ $= L/_{\sim_L}$.

Lemma 7 *Let $(S, \mathcal{A}, \mathcal{E})$ and $(S', \mathcal{A}', \mathcal{E}')$ be positively typed monoids such that $(S', \mathcal{A}', \mathcal{E}')$ divides $(S, \mathcal{A}, \mathcal{E})$. Then every language recognized by $(S', \mathcal{A}', \mathcal{E}')$ is also recognized by $(S, \mathcal{A}, \mathcal{E})$.*

Proof Since $(S', \mathcal{A}', \mathcal{E}')$ divides $(S, \mathcal{A}, \mathcal{E})$, there exist a submonoid $(T, I, \mathcal{F})$ of $(S, \mathcal{A}, \mathcal{E})$ and a surjective positively typed morphism $\beta : (T, \mathcal{I}, \mathcal{F}) \longrightarrow (S', \mathcal{A}', \mathcal{E}')$. Let $L \subseteq \Sigma^*$ be a language recognized by the positively typed monoid $(S', \mathcal{A}', \mathcal{E}')$. By definition there is a positively typed morphism $\alpha : (\Sigma^*, L, \Sigma) \longrightarrow (S', \mathcal{A}', \mathcal{E}')$.
On the other hand, we have that (Σ^*, L, Σ) is a free typed monoid and $\beta : (T, I, \mathcal{F}) \longrightarrow (S', \mathcal{A}', \mathcal{E}')$ is a surjective morphism. Hence, by Lemma 5, there exists a positively typed morphism $\varphi : (\Sigma^*, L, \Sigma) \longrightarrow (T, I, \mathcal{F})$, such that $\alpha = \beta \circ \varphi$. Then we can define a positively typed morphism $h : (\Sigma^*, L, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ as the composition $i \circ \varphi$, where i is an injective positively typed morphism from $(T, I, \mathcal{F})$ on $(S, \mathcal{A}, \mathcal{E})$.



Hence the language $L \subseteq \Sigma^*$ is also recognized by $(S, \mathcal{A}, \mathcal{E})$.

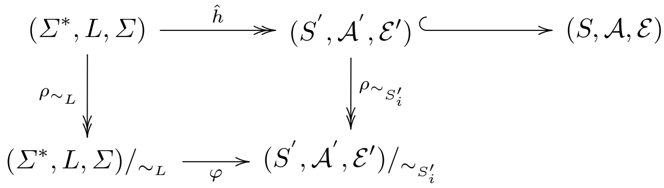
Lemma 8 *Let $L \subseteq \Sigma^*$ be a language and let $(S, \mathcal{A}, \mathcal{E})$ be a positively typed monoid. Then:*

1) *$(S, \mathcal{A}, \mathcal{E})$ recognizes L if and only if $syn(L)$ divides $(S, \mathcal{A}, \mathcal{E})$.*
2) *If $(S, \mathcal{A}, \mathcal{E})$ recognizes L and $(S, \mathcal{A}, \mathcal{E})$ divides $syn(L)$, then $(S, \mathcal{A}, \mathcal{E})$ is isomorphic to $syn(L)$.*

Proof 1) From Lemma 7 it is clear that if $syn(L)$ divides $(S, \mathcal{A}, \mathcal{E})$, then $(S, \mathcal{A}, \mathcal{E})$ recognizes L.

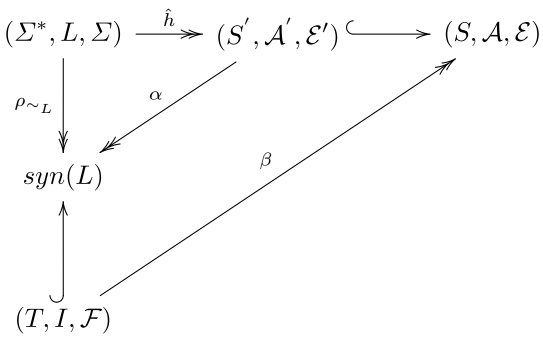
Assume now that $(S, \mathcal{A}, \mathcal{E})$ recognizes L and let $h : (\Sigma^*, L, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ be a positively typed morphism and S_i a type in $\mathcal{A}$ such that $h_m^{-1}(S_i) = L$.

Now, let $(S^{'}, \mathcal{A}^{'}, \mathcal{E}')$ be the image of h, and $\hat{h} : (\Sigma^*, L, \Sigma) \longrightarrow (S^{'}, \mathcal{A}^{'}, \mathcal{E}')$ the corresponding surjective positively typed morphism from (Σ^*, L, Σ) to $(S^{'}, \mathcal{A}^{'}, \mathcal{E}')$. Denote $S_i' = S_i \cap h_m(S)$, a type in $(S^{'}, \mathcal{A}^{'}, \mathcal{E}')$, and observe that $h_m^{-1}(S_i') = h_m^{-1}(S_i) = L$. Then by Lemma 6 there is an isomorphism $\varphi : syn(L) \longrightarrow (S^{'}, \mathcal{A}^{'}, \mathcal{E}')/\sim_{S_i'}$. Hence the composition $\alpha := \varphi^{-1} \circ \rho_{\sim_{S_i'}}$ (where $\rho_{\sim_{S_i'}}$ is the corresponding canonical projection) is a surjective positively typed morphism from $(S^{'}, \mathcal{A}^{'}, \mathcal{E}')$ to $syn(L) = (\Sigma^*, L, \Sigma)/_{\sim_L}$. This means that $syn(L)$ divides $(S, \mathcal{A}, \mathcal{E})$.



where the respective ρ's are the canonical projections.

2) Let $(S, \mathcal{A}, \mathcal{E})$ be a positively typed monoid which recognizes a language $L \subseteq \Sigma^*$ via a positively typed morphism h, and assume that $(S, \mathcal{A}, \mathcal{E})$ divides $syn(L)$. Then there exist both a submonoid $(T, I, \mathcal{F})$ of $syn(L)$ and a surjective positively typed morphism β from $(T, I, \mathcal{F})$ onto $(S, \mathcal{A}, \mathcal{E})$. Moreover, as in the preceding paragraph, we can construct a surjective positively typed morphism, say α, from $(S', A', \mathcal{E}') = h((S, \mathcal{A}, \mathcal{E}))$ (submonoid of $(S, \mathcal{A}, \mathcal{E})$) onto $syn(L)$. Hence we have the following diagram:



Since Σ generates Σ^* and $\hat{h}, \rho_L$ are surjective morphisms, $\mathcal{E}'$ generates S', and $\Sigma^*/_{\sim_L}$ is generated by $\Sigma/_{\sim_L}$. On the other hand, we have that $\mid \mathcal{E}' \mid \leq \mid \mathcal{E} \mid \leq \mid \mathcal{F} \mid \leq \mid \Sigma/_{\sim_L} \mid \leq \mid \mathcal{E}' \mid$, and hence $\mid \mathcal{E}' \mid = \mid \mathcal{E} \mid = \mid \mathcal{F} \mid = \mid \Sigma/_{\sim_L} \mid$. Now, since $\Sigma^*/_{\sim_L}$ is a finitely generated semigroup with generating set $\Sigma/_{\sim_L}$, and $(T, I, \mathcal{F})$ is a submonoid of $syn(L)$ with $\mid \mathcal{F} \mid = \mid \Sigma/_{\sim_L} \mid$, this means that $(T, \mathcal{I}, \mathcal{F})$ is isomorphic to $syn(L)$. Therefore, because of the surjectivity of β, we can deduce that $\mathcal{E}$ generates S. This latter fact implies again that $(S, \mathcal{A}, \mathcal{E})$ is isomorphic to $(S', \mathcal{A}', \mathcal{E}')$.

We have then a surjective positively typed morphism from $(S, \mathcal{A}, \mathcal{E})$ to $syn(L)$, say $\hat{\alpha}$, besides the surjective positively typed morphism β from $syn(L)$ onto

$(S,\mathcal{A},\mathcal{E})$. Since $\hat{\alpha}\circ\beta$ is a permutation of the finite set $\mathcal{E}$, then there exists some power of $\hat{\alpha}\circ\beta$ which is the identity on $\mathcal{E}$. Hence this power also defines an identity on $(S,\mathcal{A},\mathcal{E})$, and so $(S,\mathcal{A},\mathcal{E})$ is isomorphic to $syn(L)$.

Note that the fact that we are dealing with finitely generated monoids and the role of units are both relevant in the proof of the "uniqueness" of the syntactic positively typed monoid (Lemma 8 b)).

5 Positive weakly closed classes

Eilenberg introduced in [9, Chapter III] classes of transformation semigroups with weaker closure properties than varieties, the so called weakly closed classes. Also many sets of languages described via logic do not form a variety, although they enjoy weaker closure properties. Motivated by these facts, in [2] weakly closed classes of typed monoids and languages were defined, in order to obtain a weaker version of Eilenberg's variety theorem. In this section we will develop a corresponding theory in the "positive" world.

We introduce next the operations on positively typed monoids we will make use of. The first one allows to identify positively typed monoids recognizing the same languages.

Definition 7 Let $(S,\mathcal{A},\mathcal{E}),(T,\mathcal{I},\mathcal{F})$ be positively typed monoids, such that there is a surjective positively typed morphism from $(T,\mathcal{I},\mathcal{F})$ to $(S,\mathcal{A},\mathcal{E})$, then we say that $(T,\mathcal{I},\mathcal{F})$ is a *trivial extension* of $(S,\mathcal{A},\mathcal{E})$.

Observe that, in particular, if $(T,\mathcal{I},\mathcal{F})$ is a trivial extension of $(S,\mathcal{A},\mathcal{E})$, then $(S,\mathcal{A},\mathcal{E})$ divides $(T,\mathcal{I},\mathcal{F})$.

Lemma 9 *If $(T,\mathcal{I},\mathcal{F})$ is a trivial extension of $(S,\mathcal{A},\mathcal{E})$, then they recognize exactly the same languages.*

Proof From Lemma 7, it is clear that every language recognized by $(S,\mathcal{A},\mathcal{E})$ is also recognized by $(T,\mathcal{I},\mathcal{F})$. Now let L be a language recognized by $(T,\mathcal{I},\mathcal{F})$ via the positively typed morphism $h:(\Sigma^*,L,\Sigma)\longrightarrow(T,\mathcal{I},\mathcal{F})$ and let $f:(T,\mathcal{I},\mathcal{F})\longrightarrow(S,\mathcal{A},\mathcal{E})$ be a surjective positively typed morphism. Then L is recognized by $(S,\mathcal{A},\mathcal{E})$ via the composition $f\circ h$.

Another usual operation on algebraic structures is the direct product. Now we introduce it for positively typed monoids in the expected way:

Definition 8 The direct product of two positively typed monoids $(S,\mathcal{A},\mathcal{E})$ and $(T,\mathcal{I},\mathcal{F})$, denoted by $(S,\mathcal{A},\mathcal{E})\times(T,\mathcal{I},\mathcal{F})$, is defined as $(S\times T,\mathcal{A}\times\mathcal{I},\mathcal{E}\times\mathcal{F})$, where $\mathcal{A}\times\mathcal{I}$ is the positive Boolean algebra over $S\times T$ generated by the sets $A\times I$, with $A\in\mathcal{A}$, $I\in\mathcal{I}$.

We see next that direct products correspond to unions and intersections in the language counterpart.

Lemma 10 *Let $L_1, L_2 \subseteq \Sigma^*$ be languages recognized by $(S, \mathcal{A}, \mathcal{E})$ and $(T, \mathcal{I}, \mathcal{F})$ respectively. Then $L_1 \cap L_2$ and $L_1 \cup L_2$ are recognized by $(S \times T, \mathcal{A} \times \mathcal{I}, \mathcal{E} \times \mathcal{F})$.*

Proof Let $L_1, L_2 \subseteq \Sigma^*$ be recognized by $(S, \mathcal{A}, \mathcal{E})$ and $(T, \mathcal{I}, \mathcal{F})$ via the positively typed morphisms $h : (\Sigma^*, L_1, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E})$ and $h' : (\Sigma^*, L_2, \Sigma) \longrightarrow (T, \mathcal{I}, \mathcal{F})$, respectively. Let $S_i \in \mathcal{A}$, $T_i \in I$ such that $L_1 = h_m^{-1}(S_i)$ and $L_2 = h_m'^{-1}(T_i)$. Now we define a monoid morphism $\hat{h}_m : \Sigma^* \longrightarrow S \times T$ by $\hat{h}_m(x) = (h_m(x), h'_m(x))$. It is easy to prove that $\hat{h}_m^{-1}(S_i \times T_i) = L_1 \cap L_2$ and $\hat{h}_m^{-1}((S_i \times T) \cup (S \times T_i)) = L_1 \cup L_2$.

Now, by the definition of a direct product of positively typed monoids, $S_i \times T_i$ and $(S_i \times T) \cup (S \times T_i)$ are types in $(S, \mathcal{A}, \mathcal{E}) \times (T, \mathcal{I}, \mathcal{F})$. Therefore $\hat{h}$ defines positively typed morphisms $(\Sigma^*, L_1 \cap L_2, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E}) \times (T, \mathcal{I}, \mathcal{F})$ and $(\Sigma^*, L_1 \cup L_2, \Sigma) \longrightarrow (S, \mathcal{A}, \mathcal{E}) \times (T, \mathcal{I}, \mathcal{F})$. Hence $(S, \mathcal{A}, \mathcal{E}) \times (T, \mathcal{I}, \mathcal{F})$ recognizes $L_1 \cap L_2$ and $L_1 \cup L_2$, as desired.

Definition 9 A *positive weakly closed class* of languages is an operator $\mathcal{V}$ which associates to each alphabet Σ a nonempty set $\Sigma^*\mathcal{V}$ of languages such that:

1) For every alphabet Σ, $\Sigma^*\mathcal{V}$ is closed under finite union and finite intersection.
2) For every length preserving morphism $\varphi : \Sigma^* \longrightarrow \Gamma^*$ and every language $L \in \Gamma^*\mathcal{V}$, $\varphi^{-1}(L) \in \Sigma^*\mathcal{V}$.

Definition 10 A *weakly closed class* of positively typed monoids is a nonempty set of positively typed monoids that is closed under trivial extensions, division and finite direct products.

Given a nonempty set $\mathbf{V}$ of positively typed monoids, let $\mathcal{L}(\mathbf{V})$ be the mapping which associates with every alphabet Σ the nonempty set of all languages over Σ that can be recognized by a positively typed monoid in $\mathbf{V}$:

$$\Sigma^*\mathcal{L}(\mathbf{V}) = \{L \subseteq \Sigma^* \mid L \text{ is recognized by a positively typed monoid in } \mathbf{V}\}.$$

Clearly, the following result holds:

Lemma 11 *Let $\mathbf{V}$ and $\mathbf{W}$ be sets of positively typed monoids.*

1) *If $\mathbf{V}$ is closed under division, then $\Sigma^*\mathcal{L}(\mathbf{V})$ is the set of all languages $L \subseteq \Sigma^*$ with $syn(L) \in \mathbf{V}$.*
2) *If $\mathbf{V} \subseteq \mathbf{W}$, then $\Sigma^*\mathcal{L}(\mathbf{V}) \subseteq \Sigma^*\mathcal{L}(\mathbf{W})$, for every alphabet Σ.*

The next theorem ensures that there is a correspondence between positive weakly closed classes of languages and weakly closed classes of positively typed monoids. However, this correspondence is not one-to-one, in general, since for a given positive weakly closed class of languages there could be several associated weakly closed classes of positively typed monoids. Note that the proof of this theorem uses the same arguments as those in [2] for the typed case. Nevertheless, we include the proof here for the sake of comprehensiveness.

Theorem 1 1) *If* $\mathbf{V}$ *is a weakly closed class of positively typed monoids then* $\mathcal{L}(\mathbf{V})$ *is a positive weakly closed class of languages.*
2) *If* $\mathcal{V}$ *is a positive weakly closed class of languages, then there is a weakly closed class of positively typed monoids* $\mathbf{V}$ *with* $\mathcal{L}(\mathbf{V}) = \mathcal{V}$.

Proof 1) Let $\mathbf{V}$ be a weakly closed class of positively typed monoids. We are going to show that $\mathcal{V} = \mathcal{L}(\mathbf{V})$ is a positive weakly closed class of languages. Let $L \subseteq \Sigma^*$ be a language in $\Sigma^*\mathcal{V}$ and $(S, \mathcal{A}, \mathcal{E})$ be a positively typed monoid recognizing L via the positively typed morphism h. Now, assume that $L' \subseteq \Gamma^*$ is a language such that $L' = \varphi^{-1}(L)$ where $\varphi : \Gamma^* \longrightarrow \Sigma^*$ is a length preserving morphism. Since φ is length preserving, $\varphi(\Gamma) \subseteq \Sigma$, and so it can be seen as a positively typed morphism from (Γ^*, L', Γ) to (Σ^*, L, Σ). Hence $h \circ \varphi$ is a positively typed morphism from (Γ^*, L', Γ) to $(S, \mathcal{A}, \mathcal{E})$. Therefore $L' \in \Gamma^*\mathcal{V}$. This shows that $\mathcal{V}$ satisfies the closure under inverse length preserving morphisms. Since $\mathbf{V}$ is closed under finite direct products, the closure under union and intersection follows from Lemma 10.

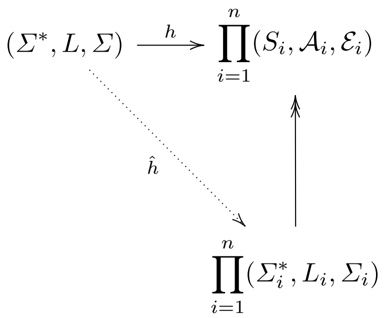
2) Let $\mathcal{V}$ be a positive weakly closed class of languages and let $\mathbf{V}$ be the smallest weakly closed class of positively typed monoids that contains all syntactic monoids of $\mathcal{V}$. Given any alphabet Σ, it is clear that $\Sigma^*\mathcal{V} \subseteq \Sigma^*\mathcal{L}(\mathbf{V})$. So we have to show that $\Sigma^*\mathcal{L}(\mathbf{V}) \subseteq \Sigma^*\mathcal{V}$.

Let L be a language in $\Sigma^*\mathcal{L}(\mathbf{V})$. To show that $L \in \Sigma^*\mathcal{V}$ we will construct a language $L' \in \Delta^*$, for some alphabet Δ, as a positive Boolean combination of languages $L_i \in \Sigma_i^*\mathcal{V}$, arising from the the positively typed monoid recognizing L, and then construct a length preserving morphism $\phi : \Sigma^* \longrightarrow \Delta^*$ such that $L = \phi^{-1}(L')$.

Since $L \in \Sigma^*\mathcal{L}(\mathbf{V})$, it is recognized by a positively typed monoid in $\mathbf{V}$. Note that division and trivial extensions do not provide more languages in $\mathcal{L}(\mathbf{V})$, because of Lemmas 8 and 9. Hence we may assume that there exist some $(S_i, \mathcal{A}_i, \mathcal{E}_i)$, which are syntactic monoids of some languages $L_i \in \Sigma_i^*\mathcal{V}$, for $i = 1, \ldots, n$, such that L is recognized via a positively typed morphism $h : (\Sigma^*, L, \Sigma) \longrightarrow \prod_{i=1}^n (S_i, \mathcal{A}_i, \mathcal{E}_i)$. Note, in particular, that every $(S_i, \mathcal{A}_i, \mathcal{E}_i) \in \mathbf{V}$.

Moreover, each language L_i is recognized via the syntactic morphism $\rho_i : (\Sigma_i^*, L_i, \Sigma_i) \longrightarrow (S_i, \mathcal{A}_i, \mathcal{E}_i)$. We can then construct a surjective positively typed morphism from $\prod_{i=1}^n (\Sigma_i^*, L_i, \Sigma_i)$ to $\prod_{i=1}^n (S_i, \mathcal{A}_i, \mathcal{E}_i)$. In particular, this means that the positively typed monoid $\prod_{i=1}^n (\Sigma_i^*, L_i, \Sigma_i)$ is a trivial extension of $\prod_{i=1}^n (S_i, \mathcal{A}_i, \mathcal{E}_i)$. By Lemma 9, the language L is also recognized by $\prod_{i=1}^n (\Sigma_i^*, L_i, \Sigma_i)$, and so there exists a positively typed morphism $\hat{h} : (\Sigma^*, L, \Sigma) \longrightarrow \prod_{i=1}^n (\Sigma_i^*, L_i, \Sigma_i)$, as shown in the diagram below.

This implies that $L = \hat{h}^{-1}(T)$ for some type $T \in \prod_{i=1}^n (\Sigma_i^*, L_i, \Sigma_i)$, so T is a Boolean combination of types $T_i \in \{\emptyset, \Sigma_i^*, L_i\}$. Now, observe that since $\hat{h}_u(\Sigma) \subseteq \prod_{i=1}^n \Sigma_i$, it can be deduced that $\hat{h}_m(\Sigma^*) \subseteq (\prod_{i=1}^n \Sigma_i)^*$. Then $\hat{h}_m : \Sigma^* \longrightarrow (\prod_{i=1}^n \Sigma_i)^*$ is a length preserving monoid morphism.



If we consider $\Delta := \prod_{i=1}^{n} \Sigma_i$, $L' = T$ and $\phi = \hat{h}$, then we have $L' \in \Delta^*\mathcal{V}$ and $L = \phi^{-1}(L')$, being ϕ a length preserving morphism. This implies that $L \in \Sigma^*\mathcal{V}$ and the result follows.

6 Positive varieties

Our aim in this section is to obtain an Eilenberg-like theorem for positive varieties of languages and varieties of positively typed monoids. The notion of positive variety of languages corresponds to the one introduced by Pin in [14]. We recall first the concepts of right and left quotients of languages:

Definition 11 The *right quotient* of a language $L \subseteq \Sigma^*$ by $a \in \Sigma^*$ is defined by $La^{-1} = \{x \in \Sigma^* \mid xa \in L\}$. The *left quotient* of a language $L \subseteq \Sigma^*$ by $a \in \Sigma^*$ is defined by $a^{-1}L = \{x \in \Sigma^* \mid ax \in L\}$.

Definition 12 (see [14]) A *positive variety of languages* is an operator $\mathcal{V}$ which associates to each alphabet Σ a nonempty set $\Sigma^*\mathcal{V}$ of languages such that:

1) For every alphabet Σ, $\Sigma^*\mathcal{V}$ is closed under finite union and finite intersection, i.e., $\Sigma^*\mathcal{V}$ is a positive Boolean algebra.
2) If $\varphi : \Sigma^* \longrightarrow \Gamma^*$ is a monoid morphism and $L \in \Gamma^*\mathcal{V}$, then $\varphi^{-1}(L) \in \Sigma^*\mathcal{V}$.
3) If $L \in \Sigma^*\mathcal{V}$ and $a \in \Sigma$, then $a^{-1}L$ and La^{-1} belong to $\Sigma^*\mathcal{V}$.

In particular, $\Sigma^*\mathcal{V}$ always contains the empty set and the whole Σ^*, since it is a positive Boolean algebra.

Note that a positive variety of languages is a positive weakly closed class of languages which is closed under quotients, and also closed for inverse morphisms which are not necessarily length preserving.

For the algebraic counterpart, new operations on positively typed monoids should be introduced, as it was done in [2]. The first one models the closure under right and left quotients of languages.

Definition 13 Let $(S, \mathcal{A}, \mathcal{E})$ be a positively typed monoid. Then $(S, \mathcal{A}', \mathcal{E})$ is a *shift* of $(S, \mathcal{A}, \mathcal{E})$, if there exist $p, q \in S$ with $\mathcal{A}' = \{p^{-1}S_iq^{-1} \mid S_i \in \mathcal{A}\}$, where $p^{-1}S_iq^{-1} = \{s \in S \mid psq \in S_i\}$.

In order to handle non-length-preserving morphisms we do not need to control the units, so the next concept is used:

Definition 14 Let $(S, \mathcal{A}, \mathcal{E})$ be a positively typed monoid. If $\mathcal{E}'$ is any finite set $\mathcal{E}' \subseteq S$, we say that $(S, \mathcal{A}, \mathcal{E}')$ is a *unit relaxation* of $(S, \mathcal{A}, \mathcal{E})$.

Now we have the new closure properties needed to introduce the notion of variety of positively typed monoids.

Definition 15 A *variety of positively typed monoids* is a weakly closed class of positively typed monoids, which is closed under shifting and unit relaxation.

Remark 4 We have noted that given a finite ordered monoid $(S, \leq)$, we can always construct a positively typed monoid $(S, \mathcal{A}, S)$, by considering $\mathcal{A}$ to be the set of order ideals of $(S, \leq)$, in the sense of Pin [14]. However, a variety of finite ordered monoids as defined in [14] does not form a variety of positively typed monoids, since it is not closed under trivial extension leading to infinite structures.

Our Eilenberg-type theorem for the positively typed world (Theorem 2) will be now deduced from the next propositions. Once proper adjustments in the definitions and previous lemmas have been done, their proofs follow the same lines as the corresponding ones for the typed case ([2, Propositions 3, 4, 5]), using the results on positive weakly closed classes in Section 5.

Proposition 1 *If* $\mathbf{V}$ *is a variety of positively typed monoids, then* $\mathcal{L}(\mathbf{V})$ *is a positive variety of languages.*

Proposition 2 *Let* $\mathbf{V}$ *and* $\mathbf{W}$ *be varieties of positively typed monoids. Then* $\mathcal{L}(\mathbf{V}) = \mathcal{L}(\mathbf{W})$ *if and only if* $\mathbf{V} = \mathbf{W}$.

Proposition 3 *For a positive variety of languages* $\mathcal{V}$, *let* $\mathbf{V}$ *be the smallest variety of positively typed monoids that contains all syntactic monoids of* $\mathcal{V}$. *Then* $\mathcal{L}(\mathbf{V}) = \mathcal{V}$.

Theorem 2 *There is a one-to-one correspondence between varieties of positively typed monoids and positive varieties of languages:*

- *Given a positive variety of languages* $\mathcal{V}$, *let* $\mathbf{V}$ *be the smallest variety of positively typed monoids that recognizes all languages in* $\mathcal{V}$. *Then* $\mathcal{L}(\mathbf{V}) = \mathcal{V}$.
- *Given a variety of positively typed monoids* $\mathbf{V}$, *let* $\mathbf{W}$ *be the smallest variety that recognizes all languages of* $\mathcal{L}(\mathbf{V})$. *Then* $\mathbf{V} = \mathbf{W}$.

7 An example and further work

We describe here an example of the correspondence between weakly closed classes of positively typed monoids and positive weakly closed class of languages, in order to initially illustrate some of the theoretical results presented in the paper. First we consider the non-regular language

$$L_a = \{a^p \mid p \text{ is prime}\}.$$

Let $\mathbb{P} = \{p \in \mathbb{Z}^+ \mid p \text{ is a prime}\}$. A positively typed monoid recognizing L_a is $V_p = (\mathbb{Z}_0^+, \{\mathbb{Z}_0^+, \mathbb{P}, \emptyset\}, \{1\})$, via the monoid morphism $h : \{a\}^* \longrightarrow \mathbb{Z}_0^+$ such that $h(a) = 1$.

Example 6 Let $\mathbf{V}$ be the smallest weakly closed class of positively typed monoids that contains V_p. Then the positive weakly closed class corresponding to $\mathbf{V}$, $\mathcal{L}(\mathbf{V})$, is the class such that for any alphabet Σ, $\Sigma^*\mathcal{L}(\mathbf{V})$ is the positive Boolean algebra generated by languages of the form $L(\Sigma) = \{x \in \Sigma^* \mid |x| \text{ is prime}\}$.

First notice that the positively typed monoid V_p recognizes all languages of the form $L(\Sigma)$, for any alphabet Σ. To see this it is enough to consider the monoid morphism ψ from Σ^* to $\{a\}^*$ such that $\psi(c) = a$ for any $c \in \Sigma$. Then V_p recognizes the language $L(\Sigma)$ via $h \circ \psi$. This is to say that $L(\Sigma) \in \Sigma^*\mathcal{L}(\{V_p\}) \subseteq \Sigma^*\mathcal{L}(\mathbf{V})$, for any Σ. Note also that $L(\Sigma) = \psi^{-1}(L_a)$, for any Σ. Now, by using Theorem 1, it suffices to prove that the operator which assigns to each alphabet Σ the positive algebra generated by languages of the form $L(\Sigma)$ is a positive weakly closed class of languages. It can be deduced from the fact that if $\varphi : \Gamma^* \longrightarrow \Sigma^*$ is a length preserving morphism, then $\varphi^{-1}(L(\Sigma)) = L(\Gamma)$, and for any languages $L, M \subseteq \Sigma^*$, $\varphi^{-1}(L) \cup \varphi^{-1}(M) = \varphi^{-1}(L \cup M)$ and $\varphi^{-1}(L) \cap \varphi^{-1}(M) = \varphi^{-1}(L \cap M)$.

Further work

In this paper we have extended the theory of typed monoids developed by Behle, Krebs and Reifferscheid in [2] to the positive case introduced by Pin in [14]. As already stated in [2], the typed approach helps to obtain a more precise description of language classes than with the usual one with monoids or with stamps as in [15]. Our extension will allow to deal with classes of non-regular languages which are not closed under complement, such as context-free languages. Further work should be developed in this sense to provide for significant examples to our positive theory.

Since the publication of [2], the theory of typed monoids has been used to give algebraic characterizations of some concrete classes of non-regular languages. Recently, Cadilhac, Krebs and McKenzie in [3] have described, by using this algebraic object, languages recognized by some variants (deterministic, unambiguous and affine) of the so-called Parikh automata. These automata were introduced in [10] and their properties were successfully used in the model-checking of hardware circuits, which indicates that they are of interest for real-world applications. As it is pointed out in [3], the class of languages defined by general Parikh automata (also called “constrained automata”) is not closed under complement, and hence this class does not allow a characterization through typed monoids. Our theory of positively typed monoids could give the suitable framework for an algebraic characterization of such classes. This is a relevant task to be addressed in the future.

Even in the regular case, using positively typed monoids one would be able to deal with some classes of languages which cannot be characterized only in terms of varieties of finite monoids. For example, classes of regular languages defined by existential formulas in a first order logic enriched by modular numerical predicates, which were characterized in [6] using C-varieties, could be potential candidates for this treatment.

A possible future research direction would be to connect our work to the category-theoretic approach recently given by Urbat, Admek, Chen and Milius in [22], or by Salamanca in [18]. At this point, our results are not direct instances of their theory. It is to be analyzed if their methods could be extended to obtain the results in this paper and those of [2].

As pointed out by one of the referees, it might be interesting to look at the class of recursively enumerable languages in the context of our work. As she/he indicates, if one could find a description of the corresponding variety of positively typed monoids, this would potentially result in a new algebraic approach to computability.

Finally, as mentioned in the introduction, Polák in [16] studied classes of languages with even weaker closure properties, the so-called conjunctive and disjunctive varieties. Therefore it is natural to wonder to what extent our results can be adapted for those conjunctive and disjunctive cases.

Acknowledgement

We thank the anonymous referees for their careful reading of the paper and for many helpful comments and suggestions which have certainly improved the final version.

References

1. Ballester-Bolinches, A., Pin, J. É., and Soler-Escrivà, X.: Formations of finite monoids and formal languages: Eilenberg's variety theorem revisited. Forum Math. **26** (6), 1737-1761 (2014)
2. Behle, C., Krebs, A., and Reifferscheid, S.: Typed monoids – An Eilenberg-like theorem for non regular languages. In: Proceedings CAI 2011, Lecture Notes in Computer Science **6742**, pp. 97–114 (2011)
3. Cadilhac, M., Krebs, A., McKenzie, P.: The algebraic theory of Parikh automata. Theory Comput. Syst., **62**, 1241–1268 (2018)
4. Cano, A., Jurvanen, E.: Varieties of languages and frontier check. In: Proceedings of 13th International Conference on Automata and Formal Languages AFL2011, pp. 153–167 (2011)
5. Cano Gómez, A., Steinby, M.: Generalized contexts and n-ary syntactic semigroups of tree languages. Asian-Eur. J. Math., **4**, 49–79 (2011)
6. Chaubard, L., Pin, J. É., Straubing, H.: First order formulas with modular predicates. In: Proceedings of the 21st Annual IEEE Symposium on Logic in Computer Science (LICS 2006), pp. 211–220, IEEE, 2006.
7. Cano Gómez, A.: Semigroupes ordonnés et opérations sur les langages rationnels, Ph.D. Thesis, Université Paris 7 and Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, (2003)
8. Cano Gómez, A., Pin, J. É.: Shuffle on positive varieties of languages. Theor. Comput. Sci., **312**, 433–461 (2004)
9. Eilenberg, S.: Automata, Languages and Machines, vol. B. Academic Press, New York, (1976)

10. Klaedtke, F., Rueß, H.: Monadic second-order logics with cardinalities. In: Baeten J.C.M., Lenstra J.K., Parrow J., Woeginger G.J. (eds), Automata, Languages and Programming. ICALP 2003, Lect. Notes Comput. Sci. **2719**, pp. 681–696, Springer, Heidelberg (2003)
11. Krebs, A., Lange, K.J., Reifferscheid, S.: Characterizing TC^0 in terms of infinite groups. Theory Comput. Syst. **40**(4), 303–325 (2007)
12. Pin, J. É.: Syntactic semigroups. In: Rozenberg, G., Salomaa, A. (Eds.), Handbook of formal languages, vol. 1 (chap. 10). Springer, Berlin, (1997)
13. Pin, J. É.: Varieties of formal languages. Plenum Pub. Corp., New York, (1986)
14. Pin, J. É.: A variety theorem without complementation. Russian Math. (Iz. VUZ) **39**, 74–83 (1995)
15. Pin, J. É., Straubing, H.: Some results on C-varieties. Theoret. Informatics Appl., **39**, 239–262 (2005)
16. Polák, L.: A classification of rational languages by semilattice-ordered monoids. Arch. Math. (Brno) **40**, 395–406 (2004)
17. Sakarovitch, J.: An algebraic framework for the study of the syntactic monoids application to the group languages. In: Mazurkiewic, A. (ed.), MFCS, pp. 510–516, Springer, Heidelberg (1976)
18. Salamanca, J.: Unveiling Eilenberg-type Correspondences: Birkhoff's Theorem for (finite) Algebras + Duality. https://arxiv.org/abs/1702.02822. February 2017.
19. Steinby, M.: A theory of tree language varieties. In: Nivat, M., Podelski, A. (eds.), Tree Automata and Languages, pp. 57–81, North-Holland, Amsterdam, (1992)
20. Straubing, H.: Finite Automata, Formal Logic, and Circuit Complexity. Birkh'̈auser, Boston, (1994)
21. Straubing, H.: On logical descriptions of regular languages. In: Rajsbaum, S. (ed.). LATIN 2002. Lect. Notes Comput. Sci. **2286**, pp. 528–538, Springer, Berlin (2002)
22. Urbat, H., Admek, J., Chen, L., Milius, S.: Eilenberg theorems for free. In: Larsen K. M., Bodlaender, H.L., Raskin, J.F. (eds), MFCS 2017. vol. **83**. pp. 43:1–43:15, LIPIcs, Leibnitz (2017) (https://arxiv.org/abs/1602. 05831. February 2017.)