Closure Properties of Pattern Languages

Joel D. Day[1], Daniel Reidenbach[1], and Markus L. Schmid[2]

[1] Department of Computer Science, Loughborough University,
Loughborough, Leicestershire, LE11 3TU, UK
J.Day@lboro.ac.uk,
D.Reidenbach@lboro.ac.uk
[2] Universität Trier, FB IV–Abteilung Informatikwissenschaften,
D-54286 Trier, Germany
MSchmid@uni-trier.de

Abstract. Pattern languages are a well-established class of languages that is particularly popular in algorithmic learning theory, but very little is known about their closure properties. In the present paper we establish a large number of closure properties of the terminal-free pattern languages, and we characterise when the union of two terminal-free pattern languages is again a terminal-free pattern language. We demonstrate that the equivalent question for general pattern languages is characterised differently, and that it is linked to some of the most prominent open problems for pattern languages. We also provide fundamental insights into a well-known construction of E-pattern languages as unions of NE-pattern languages, and vice versa.

Keywords: Pattern languages, Closure properties.

1 Introduction

Pattern languages were introduced by Dana Angluin [1] in order to model the algorithmic inferrability of patterns that are common to a set of words. In this context, a pattern is a sequence of variables and terminal symbols, and its language is the set of all words that can be generated from the pattern by a substitution that replaces all variables in the pattern by words of terminal symbols. Hence, more formally, a substitution is a terminal-preserving morphism, i. e., a morphism that maps every terminal symbol to itself. For example, the pattern language of the pattern $\alpha := x_1x_1\mathtt{a}x_2\mathtt{b}$, where x_1, x_2 are variables and $\mathtt{a}, \mathtt{b}$ are terminal symbols, is the set of all words that have a square as a prefix, followed by an arbitrary suffix that begins with the letter $\mathtt{a}$ and ends with the letter $\mathtt{b}$. Thus, e.g., $\mathtt{abbabbaab}$ is contained in the language of α, whereas $\mathtt{bbbaa}$ is not. It is a direct consequence of these definitions that a pattern language is either a singleton or infinite. Furthermore, it is worth noting that two basic types of pattern languages are considered in the literature, depending on whether the variables must stand for nonempty words (referred to as non erasing or NE-pattern languages) or whether they may represent the empty word (so-called extended, erasing or simply E-pattern languages).

A.M. Shur and M.V. Volkov (Eds.): DLT 2014, LNCS 8633, pp. 279–290, 2014.

While the definition of pattern languages is simple, many of their properties are known to be related to complex phenomena in combinatorics on words, such as pattern avoidability (see Jiang et al. [7]) and ambiguity of morphisms (see Reidenbach [12]). Hence, the knowledge on pattern languages is still patchy, despite recent progress mainly regarding decision problems (see, e. g., Freydenberger, Reidenbach [5], Fernau, Schmid [3], Fernau et al. [4] and Reidenbach, Schmid [13]) and the relation to the Chomsky hierarchy (see Jain et al. [6] and Reidenbach, Schmid [14]).

Establishing the closure properties of a class of formal languages is one of the most classical and fundamental research tasks in formal language theory and any respective progress normally leads to insights and techniques that yield a better understanding of the class. In the case of pattern languages, it is known since Angluin's initial work that they are not closed under most of the usual operations, including union, intersection and complement. However, these non-closure properties can be shown by using very basic example patterns and exploiting peculiarities of the definition of pattern languages. For example, if a pattern does not contain a variable, then its language is a singleton; hence the union of any two distinct singleton pattern languages contains two elements, and therefore it cannot be a pattern language. Furthermore, the intersection of two pattern languages given by patterns that start with different terminal symbols is empty and the empty set, although a trivial language, is not a pattern language as well. Since, apart from a strong result by Shinohara [15] on the union of NE-pattern languages, hardly anything is known beyond such immediate facts, we can observe that in the case of pattern languages the existing closure properties fail to contribute to our understanding of their intrinsic properties.

It is the main purpose of this paper to investigate the closure properties of pattern languages more thoroughly. To this end, in Section 3, we consider the closure properties of two important subclasses of pattern languages, namely the classes of terminal-free NE- and E-pattern languages, i. e., pattern languages that are generated by patterns that do not contain any terminal symbols. This choice is motivated by the fact that terminal-free patterns have been a recent focus of interest in the research on pattern languages and, furthermore, most existing examples for non-closure of pattern languages (including the two examples for union and intersection given in the previous paragraph) do not translate to the terminal-free case. In Section 3.1, we completely characterise when the union of two terminal-free pattern languages is again a terminal-free pattern language and, in Section 3.2, we prove their non-closure under intersection, for which the situation is much more complicated compared to the operation of union.

We consider general pattern languages in Section 4, and we provide complex examples demonstrating that it is probably a very hard task to obtain full characterisations of those pairs of pattern languages whose unions or intersections are again a pattern language. In Section 4.3, we also study the question whether an E-pattern language can be expressed by the union of *nonerasing* pattern languages and, likewise, whether an NE-pattern language can be expressed by the union of *erasing* pattern languages. This question is slightly at odds with the

classical investigation of closure properties, since we apply a language operation to members of one class and ask whether the resulting language is a member of another class. However, in the case of pattern languages, this makes sense, since every NE-pattern language is a finite union of E-pattern languages and every E-pattern language is a finite union of NE-pattern languages (see Jiang et al. [7]), a phenomenon that has been widely utilised in the context of inductive inference of pattern languages (see, e.g., Wright [17], Shinohara, Arimura [16]).

Due to space constraints, all proofs have been omitted from this paper.

2 Definitions and Preliminary Results

The symbols $\cup$, $\cap$ and $\setminus$ denote the set operations of *union*, *intersection* and *set difference*, respectively. For sets U and B with $B \subseteq U$, $\overline{B} := U \setminus B$ is the *complement* of B.

Let $\mathbb{N} := \{1, 2, 3, \ldots\}$ and let $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$. For an arbitrary alphabet A, a *word* (*over* A) is a finite sequence of symbols from A, and ε stands for the *empty word*. The notation A^+ denotes the set of all nonempty words over A, and $A^* := A^+ \cup \{\varepsilon\}$. For the *concatenation* of two words w_1, w_2 we write $w_1 \cdot w_2$ or simply $w_1 w_2$, and w^n stands for the n-fold concatenation of the word w. We say that a word $v \in A^*$ is a *factor* of a word $w \in A^*$ if there are $u_1, u_2 \in A^*$ such that $w = u_1 \cdot v \cdot u_2$. If u_1 (or u_2) is the empty word, then v is a *prefix* (or a *suffix*, respectively) of w. The notation $|K|$ stands for the size of a set K or the length of a word K. A word w is *primitive* if, for any u such that $w = u^k$, $k = 1$. The *primitive root* of a word w is the primitive word u such that $w = u^k$, $k \in \mathbb{N}$.

For any alphabets A, B, a *morphism* is a function $h : A^* \to B^*$ that satisfies $h(vw) = h(v)h(w)$ for all $v, w \in A^*$; h is said to be *nonerasing* if, for every $a \in A$, $h(a) \neq \varepsilon$. A morphism h is *ambiguous (with respect to a word w)* if there exists a morphism g satisfying $g(w) = h(w)$ and, for a letter a in w, $g(a) \neq h(a)$. If such a morphism g does not exist, then h is called *unambiguous (with respect to w)*. A morphism $\sigma : A^* \to B^*$ is *periodic* if for some (primitive) word $w \in B^*$, $\sigma(x) \in \{w\}^*$ for every $x \in A$. The word w will be referred to as the *primitive root* of σ. If $|\sigma(x)| = 1$ for every $x \in A$, then σ is *1-uniform*.

Let Σ be a finite alphabet of so-called *terminal symbols* and X a countably infinite set of *variables* with $\Sigma \cap X = \emptyset$. We normally assume $X := \{x_1, x_2, x_3, \ldots\}$. A *pattern* is a nonempty word over $\Sigma \cup X$, a *terminal-free pattern* is a nonempty word over X; if a word contains symbols from Σ only, then we occasionally call it a *terminal word*. For any pattern α, we refer to the set of variables in α as $\mathrm{var}(\alpha)$. If the variables in a pattern α are labelled in the natural way, then it is said to be in *canonical form*, i. e., α is in canonical form if, for some $n \in \mathbb{N}$, $\mathrm{var}(\alpha) = \{x_1, x_2, \ldots, x_n\}$ and, for any $x_i, x_j \in \mathrm{var}(\alpha)$ with $i < j$, there is a prefix β of α such that $x_i \in \mathrm{var}(\beta)$ and $x_j \notin \mathrm{var}(\beta)$. A pattern α is a *one-variable* pattern if $|\mathrm{var}(\alpha)| = 1$. A morphism $h : (\Sigma \cup X)^* \to (\Sigma \cup X)^*$ is *terminal-preserving* if $h(a) = a$ for every $a \in \Sigma$. The *residual* of a pattern α is the word $h_\varepsilon(\alpha)$, where $h_\varepsilon : (\Sigma \cup X)^* \to (\Sigma \cup X)^*$ is a terminal preserving morphism with $h_\varepsilon(x) := \varepsilon$ for every $x \in \mathrm{var}(\alpha)$. A terminal-preserving morphism $h : (\Sigma \cup X)^* \to \Sigma^*$ is called a *substitution*.

Definition 1. *Let Σ be an alphabet, and let $\alpha \in (\Sigma \cup X)^*$ be a pattern. The* E-pattern language *of α is defined by $L_{\mathrm{E},\Sigma}(\alpha) := \{h(\alpha) \mid h : (\Sigma \cup X)^* \to \Sigma^* \text{ is a substitution}\}$. The* NE-pattern language *of α is defined by $L_{\mathrm{NE},\Sigma}(\alpha) := \{h(\alpha) \mid h : (\Sigma \cup X)^* \to \Sigma^* \text{ is a nonerasing substitution}\}$.*

Note that we call a pattern language terminal-free if there exists a terminal-free pattern that generates it.

Some parts of our reasoning in the subsequent sections is based on word equations, which are defined as follows. For a set of unknowns Y, a terminal alphabet Σ, and two words $\alpha, \beta \in (Y \cup \Sigma)^+$, the expression $\alpha = \beta$ is called a *word equation*. The solutions are terminal-preserving morphisms $\sigma : (Y \cup \Sigma)^* \to \Sigma^*$ such that $\sigma(\alpha) = \sigma(\beta)$. The words $\sigma(\alpha)$ $(= \sigma(\beta))$ will be referred to as *solution-words*. It is often convenient to interpret variables from patterns as unknowns, and so word equations will often be formulated from two patterns.

This concludes the basic definitions of this paper. We now begin our investigation of the closure properties of the class of pattern languages. As a starting point, we refer to the corresponding result in the initial paper on pattern languages:

Theorem 1 (Angluin [1]). NE*-pattern languages are not closed under union, intersection, complement, Kleene plus, homomorphism and inverse homomorphism.* NE*-pattern languages are closed under concatenation and reversal.*

3 Terminal-Free Patterns

As briefly explained in Section 1, the proof of Theorem 1 heavily relies on the fact that patterns can contain terminal symbols. In the present section, we therefore wish to study whether the situation changes if we consider the classes of terminal-free E-pattern languages and terminal-free NE-pattern languages.

3.1 Union

Simple examples show that neither the terminal-free NE-pattern languages nor the terminal-free E-pattern languages are closed under union:

Proposition 1. *Let Σ be an alphabet with $\{\mathtt{a}, \mathtt{b}\} \subseteq \Sigma$. For every $Z, Z' \in \{\mathrm{E}, \mathrm{NE}\}$, there does not exist a pattern γ, such that $L_{Z,\Sigma}(\gamma) = L_{Z',\Sigma}(x_1x_1) \cup L_{Z',\Sigma}(x_1x_1x_1)$.*

It is worth noting that the above statement also provides a first minor insight into the topic of expressing E-pattern languages as unions of NE-pattern languages and vice versa. We shall study this subject in Section 4.3 for patterns with terminal symbols in much more detail. In the present section, we merely want to point out that the union of two terminal-free E-pattern languages is indeed never a terminal-free NE-pattern language, and the union of two terminal-free NE-pattern languages cannot be a terminal-free E-pattern language:

Proposition 2. *Let Σ be an arbitrary alphabet, and let α and β be terminal-free patterns. Then there does not exist a terminal-free pattern γ with $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{NE},\Sigma}(\gamma)$ or $L_{\mathrm{NE},\Sigma}(\alpha) \cup L_{\mathrm{NE},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$.*

In the remainder of this section we wish to prove a similarly strong result for the actual closure of the class of terminal-free E- or NE-pattern languages. Hence, we wish to characterise those pairs of terminal-free (NE-/E-)pattern languages where the union again is a terminal-free (NE-/E-)pattern language. Our results shall demonstrate that the union of two terminal-free E-pattern languages can only be a terminal-free E-pattern language if there is an inclusion relation between the two languages, and that the same holds for the NE-pattern languages.

Our reasoning on the E case is based on a result on the inclusion problem for E-pattern languages. In [8], Jiang et al. provide a construction for a morphism τ_k such that, for two patterns α and β, the word $\tau_{|\beta|}(\alpha)$ is contained in $L_{\mathrm{E},\Sigma}(\beta)$ if and only if there exists a morphism φ from β to α. This in turn implies that the erasing languages of two terminal free patterns satisfy a subset relation if and only if there exists a morphism from one pattern to the other. It is not difficult to see that this construction can be further used to satisfy, for patterns α, β, γ, and $k = \max(|\alpha|, |\beta|)$, that $\tau_k(\gamma) \in L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta)$ if and only if γ is a morphic image of α or β. Thus if the relation $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$ is satisfied, then $L_{\mathrm{E},\Sigma}(\gamma)$ is a subset of (and therefore also equal to) $L_{\mathrm{E},\Sigma}(\alpha)$ or $L_{\mathrm{E},\Sigma}(\beta)$ and we have the following situation.

Lemma 1. *Let Σ be an alphabet, $|\Sigma| \geq 2$, and let α and β be terminal-free patterns. There exists a terminal-free pattern γ with $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$ if and only if $L_{\mathrm{E},\Sigma}(\alpha) \subseteq L_{\mathrm{E},\Sigma}(\beta)$ or $L_{\mathrm{E},\Sigma}(\beta) \subseteq L_{\mathrm{E},\Sigma}(\alpha)$.*

It can be observed from simple examples that, in the nonerasing case, inclusion cannot be characterised by the existence of a morphism between the generating patterns. Thus, no equivalent argument can be derived for the nonerasing case. However, a corresponding result can be obtained by looking at the shortest words in the nonerasing languages of α, β and γ. To this end, we define, for a pattern α, the set M_α to be $\{\sigma(\alpha) \mid \sigma : \mathrm{var}(\alpha)^* \to \Sigma^* \text{ is 1-uniform}\}$.

The set M_α has been used to positive effect in existing literature (see, e.g., [9]). It is particularly useful when considering nonerasing pattern languages because it encodes exactly the original pattern α (up to a renaming of variables). Moreover, it has a number of convenient properties when considering the union of two NE-pattern languages. One such example is that if α is strictly shorter than β, then the set of shortest words in $L_{\mathrm{NE},\Sigma}(\alpha) \cup L_{\mathrm{NE},\Sigma}(\beta)$ will be exactly M_α. Thus, if the union is itself the nonerasing language of some pattern γ, we have that $\gamma = \alpha$ up to a renaming of variables. A similar result can be obtained for the case that $|\alpha| = |\beta|$ by considering $|M_\alpha \cup \mathrm{M}_\beta|$.

Lemma 2. *Let Σ be an alphabet, $|\Sigma| \geq 2$, and let α, β be terminal free patterns in canonical form with $|\alpha| = |\beta|$. Suppose that γ is a terminal free pattern (again in canonical form) with $\mathrm{M}_\alpha \cup \mathrm{M}_\beta = \mathrm{M}_\gamma$. Then $\gamma \in \{\alpha, \beta\}$.*

Consequently, we can verify the same statement for nonerasing languages as we have for erasing languages.

Lemma 3. *Let Σ be an alphabet, $|\Sigma| \geq 2$, and let α and β be terminal-free patterns. There exists a terminal-free pattern γ with $L_{\mathrm{NE},\Sigma}(\alpha) \cup L_{\mathrm{NE},\Sigma}(\beta) = L_{\mathrm{NE},\Sigma}(\gamma)$ if and only if $L_{\mathrm{NE},\Sigma}(\alpha) \subseteq L_{\mathrm{NE},\Sigma}(\beta)$ or $L_{\mathrm{NE},\Sigma}(\beta) \subseteq L_{\mathrm{NE},\Sigma}(\alpha)$.*

Note that Lemma 3 extends an equivalent result by Shinohara [15] that holds for alphabets with at least 3 letters.

Thus, in general, the languages of two terminal-free patterns only union together to produce a third in the trivial case.

Theorem 2. *Let Z, $Z' \in \{\mathrm{E}, \mathrm{NE}\}$. Let Σ be an alphabet, $|\Sigma| \geq 2$, and let α, β, γ be terminal-free patterns. Then $L_{Z,\Sigma}(\alpha) \cup L_{Z,\Sigma}(\beta) = L_{Z',\Sigma}(\gamma)$ if and only if $L_{Z,\Sigma}(\alpha) = L_{Z',\Sigma}(\gamma)$ and $L_{Z,\Sigma}(\beta) \subseteq L_{Z,\Sigma}(\alpha)$ or $L_{Z,\Sigma}(\beta) = L_{Z',\Sigma}(\gamma)$ and $L_{Z,\Sigma}(\alpha) \subseteq L_{Z,\Sigma}(\beta)$.*

It is worth noting that, for terminal-free patterns, the inclusion problem – and therefore the question of closure under union – is decidable in the E case (see Jiang et al. [8], as explained above), but still open in the NE case.

3.2 Intersection

In the present section, we wish to investigate if the terminal-free NE- or E-pattern languages are closed under intersection. For the NE case, simple counterexamples such as $\alpha := xyx$ and $\beta := xxy$ can be used to prove the following observation:

Proposition 3. *The terminal-free* NE*-pattern languages are not closed under intersection.*

We can obtain an equivalent result for the terminal-free E-pattern languages, but our reasoning is significantly more complex and requires the analysis of certain word equations. Moreover, we are able to provide a characterisation for a restricted class of pairs of patterns, and show that, for this class, the situation is non-trivial (i.e., there exist both positive and negative examples). We proceed by considering the link between word equations and intersections of pattern-languages.

If, for a word equation $\alpha = \beta$, the words α and β are over disjoint alphabets, then the set of solutions $\sigma : (\mathrm{var}(\alpha) \cup \mathrm{var}(\beta))^* \to \Sigma^*$ corresponds exactly to the set of pairs of morphisms $\tau_1 : \mathrm{var}(\alpha)^* \to \Sigma^*$, $\tau_2 : \mathrm{var}(\beta)^* \to \Sigma^*$ such that $\tau_1(\alpha) = \tau_2(\beta)$. Thus, it also exactly describes the intersection $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$. Furthermore, such an intersection is invariant under renamings of α and of β, so any intersection of E-pattern languages can be described in this way. The next proposition gives a characterisation of when the intersection of two terminal-free E-pattern languages is again a terminal-free E-pattern language in the restricted case that the corresponding word equation permits only periodic solutions. Note that, for α and β over disjoint alphabets, such solutions always exist.

Proposition 4. *Let Σ be an arbitrary alphabet. Let α, β be terminal-free patterns over disjoint alphabets and suppose that the word equation $\alpha = \beta$ permits only periodic solutions. Let w be the shortest non-empty solution-word. Let*

$$\mu := \mathrm{lcm}(\gcd\{|\alpha|_{x_i} \mid x_i \in \mathrm{var}(\alpha)\},\ \gcd\{|\beta|_{y_j} \mid y_j \in \mathrm{var}(\beta)\}).$$

Then $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$ is a terminal-free E-pattern language if and only if $\mu = |w|$.

Despite Proposition 4, it is still a non-trivial task to find two terminal-free E-pattern languages whose intersection is not a terminal-free E-pattern language. In particular, it remains to find appropriate patterns α and β such that the word equation $\alpha = \beta$ has only periodic solutions. The following proposition provides such an example, and hence we have the analogous result to Proposition 3.

Proposition 5. *Let Σ be an arbitrary alphabet, and let $\alpha := x_1x_2x_1^2x_2x_1^3x_2^2$ and $\beta := x_3x_4^2x_3^2x_4^6x_3^3$. Then $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$ cannot be expressed as a terminal-free E-pattern language.*

It is even possible to give a much stronger statement, showing the extent to which the 'pattern-language mechanism' is incapable of handling this seemingly uncomplicated set of solutions.

Corollary 1. *For any alphabet Σ, $L_{\mathrm{E},\Sigma}(x_1x_2x_1^2x_2x_1^3x_2^2) \cap L_{\mathrm{E},\Sigma}(x_3x_4^2x_3^2x_4^6x_3^3)$ cannot be expressed as a finite union of terminal-free E-pattern languages.*

It is worth noting that the approach above can be used to show that for $\alpha' := x_1x_2x_1^2x_2^2x_1^3x_2^3$ and $\beta' := x_3x_4^2x_3^2x_4^7x_3^3$, one has that $L_{\mathrm{E},\Sigma}(\alpha') \cap L_{\mathrm{E},\Sigma}(\beta') = L_{\mathrm{E},\Sigma}(x_1^6)$. This demonstrates that the intersection of two E-pattern languages can in some cases be expressed as an E-pattern language, and therefore that the problem of whether the intersection of two E-pattern languages form an E-pattern language is nontrivial. However it is worth pointing out that a characterisation of this situation is probably very difficult to acquire due to the challenging nature of finding solution-sets of word equations.

3.3 Other Closure Properties

In this Section, we show that regarding the closure under the operations of complementation, morphisms, inverse morphisms, Kleene plus and Kleene star, terminal-free pattern languages behave similarly to the full class of pattern languages.

Proposition 6. *For every terminal-free pattern α, $\overline{L_{\mathrm{E},\Sigma}(\alpha)}$ is not a terminal-free* E*-pattern language and $\overline{L_{\mathrm{NE},\Sigma}(\alpha)}$ is not a terminal-free* NE*-pattern language.*

Proposition 6 does not only prove the non-closure of terminal-free E- and NE-pattern languages under complementation, but also characterises in a trivial way the terminal-free pattern languages whose complement is also a terminal-free pattern language.

Proposition 7. *Let Σ be a terminal alphabet with $|\Sigma| \geq 2$. The terminal-free* NE- *and* E*-pattern languages, with respect to Σ, are not closed under morphisms, inverse morphisms, Kleene plus and Kleene star.*

4 General Patterns

As explained in Section 1 and formally stated in Theorem 1, the closure properties of the full classes of NE-pattern languages and of E-pattern languages are understood. In the present section, we therefore wish to expand the more specific insights into the terminal-free pattern languages gained in Section 3 to the full classes. More precisely, with respect to the operations of complementation, intersection and union, we investigate those patterns that exhibit the property that their complement, intersection or union is again a pattern language and we try to characterise these patterns. Our strongest results are with respect to the operation of union.

4.1 Complement

With respect to the full class of E- and NE-pattern language, an analogue of Proposition 6 exists:

Proposition 8 (Bayer [2]). *Let Σ be a terminal alphabet with $\Sigma \geq 2$. For every pattern α, $\overline{L_{\mathrm{E},\Sigma}(\alpha)}$ is not an* E*-pattern language and $\overline{L_{\mathrm{NE},\Sigma}(\alpha)}$ is not an* NE*-pattern language.*

In the same way as Proposition 6 does for terminal-free patterns, this proposition yields a trivial characterisation of pattern languages with a complement that again is a pattern language.

4.2 Intersection

It is straightforward to construct patterns α and β such that $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$ is not an E-pattern language or $L_{\mathrm{NE},\Sigma}(\alpha) \cap L_{\mathrm{NE},\Sigma}(\beta)$ is not an NE-pattern language. Furthermore, any two terminal-free patterns α and β are an example for the situation that $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$ is not an NE-pattern language and, as long as there are at least two symbols in Σ, also for the situation that $L_{\mathrm{NE},\Sigma}(\alpha) \cap L_{\mathrm{NE},\Sigma}(\beta)$ is not an E-pattern language. Moreover, there are non-trivial examples of patterns α, β and γ, such that $L_{\mathrm{NE},\Sigma}(\alpha) \cap L_{\mathrm{NE},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$:

- $L_{\mathrm{NE},\Sigma}(\mathtt{a}x) \cap L_{\mathrm{NE},\Sigma}(xx) = L_{\mathrm{E},\Sigma}(\mathtt{a}x\mathtt{a}x)$.
- $L_{\mathrm{NE},\Sigma}(x\mathtt{a}y) \cap L_{\mathrm{NE},\Sigma}(xxx) = L_{\mathrm{E},\Sigma}(x\mathtt{a}yx\mathtt{a}yx\mathtt{a}y)$.
- $L_{\mathrm{NE},\Sigma}(\mathtt{a}x\mathtt{a}) \cap L_{\mathrm{NE},\Sigma}(xx) = L_{\mathrm{E},\Sigma}(\mathtt{a}x\mathtt{aa}x\mathtt{a})$.
- $L_{\mathrm{NE},\Sigma}(\mathtt{a}x\mathtt{a}x) \cap L_{\mathrm{NE},\Sigma}(x\mathtt{b}x\mathtt{b}) = L_{\mathrm{E},\Sigma}(x\mathtt{ba}x\mathtt{b})$.
- $L_{\mathrm{NE},\Sigma}(\mathtt{a}xy) \cap L_{\mathrm{NE},\Sigma}(xxx) = L_{\mathrm{E},\Sigma}(\mathtt{a}x\mathtt{a}x\mathtt{a}x)$.

However, it is not known whether or not there are patterns α and β, such that $L_{\mathrm{E},\Sigma}(\alpha) \cap L_{\mathrm{E},\Sigma}(\beta)$ is an NE-pattern language. Moreover, we do not have any characterisations for the situation that the intersection of two pattern languages is again a pattern language.

4.3 Union

Examples of patterns α and β such that $L_{Z,\Sigma}(\alpha) \cup L_{Z,\Sigma}(\beta)$ is not a Z'-pattern language, for all $Z, Z' \in \{\mathrm{E}, \mathrm{NE}\}$, are provided by Proposition 1.

Theorem 2 is our strongest result in Section 3, as it shows that the union of terminal-free pattern languages can only be a terminal-free pattern language if one of the languages is contained in the other. At first glance it seems a reasonable hypothesis that a similar result might hold for the full class of pattern languages, but in the present section we show that this is not true.

For all but the union of pairs of E-pattern languages and the question of whether they can form an E-pattern language, suitable examples are not too hard to find:

Proposition 9. *Let Σ be a terminal alphabet.*

- $L_{\mathrm{E},\{\mathtt{a},\mathtt{b}\}}(\mathtt{a}\mathtt{a}x) \cup L_{\mathrm{E},\{\mathtt{a},\mathtt{b}\}}(\mathtt{a}\mathtt{b}x) = L_{\mathrm{NE},\{\mathtt{a},\mathtt{b}\}}(\mathtt{a}x)$.
- $L_{\mathrm{NE},\Sigma}(\mathtt{a}\mathtt{b}\mathtt{c}) \cup L_{\mathrm{NE},\Sigma}(\mathtt{a}x\mathtt{b}x\mathtt{c}x) = L_{\mathrm{E},\Sigma}(\mathtt{a}x\mathtt{b}x\mathtt{c}x)$.
- $L_{\mathrm{NE},\{\mathtt{a},\mathtt{b}\}}(\mathtt{a}x_1) \cup L_{\mathrm{NE},\{\mathtt{a},\mathtt{b}\}}(\mathtt{b}x_1) = L_{\mathrm{NE},\{\mathtt{a},\mathtt{b}\}}(x_1x_2)$.

Regarding the question of whether $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$ for patterns α, β, γ implies that there is an inclusion relation between $L_{\mathrm{E},\Sigma}(\alpha)$ and $L_{\mathrm{E},\Sigma}(\beta)$, the following three propositions provide increasingly complex counterexamples for alphabet sizes 2, 3, and 4.

Proposition 10. *Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $\alpha := x_1\mathtt{a}x_2\mathtt{b}x_2\mathtt{a}x_3$, $\beta := x_1\mathtt{a}x_2\mathtt{b}\mathtt{b}x_2\mathtt{a}x_3$ and $\gamma := x_1\mathtt{a}x_2\mathtt{b}x_3\mathtt{a}x_4$. Then $L_{\mathrm{E},\Sigma}(\alpha) \not\subseteq L_{\mathrm{E},\Sigma}(\beta)$, $L_{\mathrm{E},\Sigma}(\beta) \not\subseteq L_{\mathrm{E},\Sigma}(\alpha)$ and $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$.*

Proposition 11. *Let $\Sigma := \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$,*

$$\begin{aligned}
\alpha &:= x_1\mathtt{a}x_2x_3^6x_4^3x_5^6x_6\mathtt{b}x_7\mathtt{a}x_2x_8^{12}x_4^6x_9^{12}x_6\mathtt{b}x_{10},\\
\beta &:= x_1\mathtt{a}x_2x_3^6x_4^2x_5^5x_6^6x_7\mathtt{b}x_8\mathtt{a}x_2x_9^{12}x_4^4x_5^{10}x_{10}^{12}x_7\mathtt{b}x_{11} \text{ and}\\
\gamma &:= x_1\mathtt{a}x_2x_3^6x_4^2x_5^3x_6^6x_7\mathtt{b}x_8\mathtt{a}x_2x_9^{12}x_4^4x_5^6x_{10}^{12}x_7\mathtt{b}x_{11}.
\end{aligned}$$

Then

$$L_{\mathrm{E},\Sigma}(\alpha) \not\subseteq L_{\mathrm{E},\Sigma}(\beta),\ L_{\mathrm{E},\Sigma}(\beta) \not\subseteq L_{\mathrm{E},\Sigma}(\alpha) \text{ and } L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma).$$

Proposition 12. *Let $\Sigma := \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$,*

$$\begin{aligned}
\alpha &:= x_1\mathtt{a}x_2x_3^2x_4^2x_5^2x_6\mathtt{b}x_7\mathtt{a}x_2x_8^2x_4^2x_9^2x_6\mathtt{b}x_{10}\mathtt{c}x_{11}x_{12}^2x_{13}^2x_{14}^2x_{15}^2x_{16}\mathtt{d}\\
&\quad x_{17}\mathtt{c}x_{11}x_{18}^2x_{13}^2x_{14}^2x_{19}^2x_{16}\mathtt{d}x_{20}x_{13}^2x_{14}^2x_{13}^2x_{14}^2x_{13}^2x_{14}^2x_{21}x_4^6,\\
\beta &:= x_1\mathtt{a}x_2x_3^2x_4^2x_5^2x_6^2x_7\mathtt{b}x_8\mathtt{a}x_2x_9^2x_4^2x_5^2x_{10}^2x_7\mathtt{b}x_{11}\mathtt{c}x_{12}x_{13}^2x_{14}^2x_{15}^2x_{16}\mathtt{d}\\
&\quad x_{17}\mathtt{c}x_{12}x_{18}^2x_{14}^2x_{19}^2x_{16}\mathtt{d}x_{20}x_{14}^6x_{21}x_4^2x_5^2x_4^2x_5^2x_4^2x_5^2 \text{ and}\\
\gamma &:= x_1\mathtt{a}x_2x_3^2x_4^2x_5^2x_6^2x_7\mathtt{b}x_8\mathtt{a}x_2x_9^2x_4^2x_5^2x_{10}^2x_7\mathtt{b}x_{11}\mathtt{c}x_{12}x_{13}^2x_{14}^2x_{15}^2x_{16}^2x_{17}\mathtt{d}\\
&\quad x_{18}\mathtt{c}x_{12}x_{19}^2x_{14}^2x_{15}^2x_{20}^2x_{17}\mathtt{d}x_{21}x_{14}^2x_{15}^2x_{14}^2x_{15}^2x_{14}^2x_{15}^2x_{22}x_4^2x_5^2x_4^2x_5^2x_4^2x_5^2.
\end{aligned}$$

Then

$$L_{\mathrm{E},\Sigma}(\alpha) \not\subseteq L_{\mathrm{E},\Sigma}(\beta),\ L_{\mathrm{E},\Sigma}(\beta) \not\subseteq L_{\mathrm{E},\Sigma}(\alpha) \text{ and } L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma).$$

We are not able to give equivalent examples for larger alphabets, and we expect the question of their existence to be a complex and important problem. This is because the above examples depend on the ambiguity of terminal-preserving morphisms, which is a phenomenon that underpins many properties of pattern languages. Similar constructions to those in Propositions 10, 11, and 12 have been used to disprove longstanding conjectures on inductive inference (see Reidenbach [10,12]) of and the equivalence problem (see Reidenbach [11]) for E-pattern languages over alphabets of up to 4 letters and, similarly, it has so far not been possible to expand those techniques to arbitrary alphabets. Our examples, thus, suggest a close link between the problem in the current section and the two most important open problems for E-pattern languages over alphabets with at least 5 letters, and we expect that substantial progress on any one of them will require combinatorial insights that will allow the others to be solved as well.

For all $Z, Z' \in \{\mathrm{E}, \mathrm{NE}\}$, we have seen example patterns α and β such that $L_{Z,\Sigma}(\alpha) \cup L_{Z,\Sigma}(\beta)$ is a Z'-pattern language. We shall now try to generalise these examples in order to obtain characterisations of such pairs of patterns.

For the case $Z = Z' = \mathrm{E}$, we are only able to state a necessary condition for $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$ that, unfortunately, is not very strong:

Theorem 3. *Let Σ be an alphabet, and let α, β and γ be patterns with $L_{\mathrm{E},\Sigma}(\alpha) \cup L_{\mathrm{E},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$. Furthermore, let w_α, w_β and w_γ be the residuals of α, β and γ, respectively. Then $w_\gamma = w_\alpha$ and w_γ is a subsequence of w_β or $w_\gamma = w_\beta$ and w_γ is a subsequence of w_α.*

In view of the fact that the examples of Propositions 10, 11 and 12 are rather complicated, we expect that a full characterisation for the case $Z = Z' = \mathrm{E}$ is difficult to obtain.

For the case $Z = Z' = \mathrm{NE}$, we can present a strong necessary condition that, similarly to Lemma 3, strengthens a result by Shinohara [15]:

Theorem 4. *Let Σ be an alphabet with $\{\mathtt{a}, \mathtt{b}\} \subseteq \Sigma$ and let α, β and γ be patterns. If $L_{\mathrm{NE},\Sigma}(\alpha) \cup L_{\mathrm{NE},\Sigma}(\beta) = L_{\mathrm{NE},\Sigma}(\gamma)$, then one of the following three statements is true:*

- *$L_{\mathrm{NE},\Sigma}(\alpha) \subseteq L_{\mathrm{NE},\Sigma}(\beta)$ and $\beta = \gamma$.*
- *$L_{\mathrm{NE},\Sigma}(\beta) \subseteq L_{\mathrm{NE},\Sigma}(\alpha)$ and $\alpha = \gamma$.*
- *$|\Sigma| = 2$ and*

$$\alpha = \delta_0\, \mathtt{a}\, \delta_1\, \mathtt{a}\, \delta_2 \ldots \delta_{m-1}\, \mathtt{a}\, \delta_m\,,$$
$$\beta = \delta_0\, \mathtt{b}\, \delta_1\, \mathtt{b}\, \delta_2 \ldots \delta_{m-1}\, \mathtt{b}\, \delta_m\,,$$
$$\gamma = \delta_0\, x\, \delta_1\, x\, \delta_2 \ldots \delta_{m-1}\, x\, \delta_m\,,$$

where $m \geq 1$, $\delta_i \in (X \cup \Sigma)^*$, $0 \leq i \leq m$.

It remains to consider the cases $Z = \mathrm{NE}$, $Z' = \mathrm{E}$ and $Z = \mathrm{E}$, $Z' = \mathrm{NE}$, for which we have full characterisations. Before we prove these characterisations, we recall that Jiang et al. show in [7] that, for every pattern α, we can construct finite

sets of patterns Γ and Δ such that $L_{\mathrm{E},\Sigma}(\alpha) = \bigcup_{\beta\in\Gamma} L_{\mathrm{NE},\Sigma}(\beta)$ and $L_{\mathrm{NE},\Sigma}(\alpha) = \bigcup_{\beta\in\Delta} L_{\mathrm{E},\Sigma}(\beta)$. More precisely, Γ is the set of all patterns that can be obtained from α by erasing some (possibly none) of the variables and Δ contains all pattern that can be obtained from α by substituting each $x \in \mathrm{var}(\alpha)$ by bx, for some $b \in \Sigma$. We note that the examples $L_{\mathrm{NE},\Sigma}(\mathtt{abc}) \cup L_{\mathrm{NE},\Sigma}(\mathtt{a}x\mathtt{b}x\mathtt{c}x) = L_{\mathrm{E},\Sigma}(\mathtt{a}x\mathtt{b}x\mathtt{c}x)$ and $L_{\mathrm{E},\{\mathtt{a},\mathtt{b}\}}(\mathtt{aa}x) \cup L_{\mathrm{E},\{\mathtt{a},\mathtt{b}\}}(\mathtt{ab}x) = L_{\mathrm{NE},\{\mathtt{a},\mathtt{b}\}}(\mathtt{a}x)$ of Proposition 9 are applications of exactly this construction.

The characterisation for the case $Z = \mathrm{NE}$, $Z' = \mathrm{E}$ follows from the fact that we can prove that if we restrict ourselves to unions of only two pattern languages, then $L_{\mathrm{E},\Sigma}(\alpha) = \bigcup_{\beta\in\Gamma} L_{\mathrm{NE},\Sigma}(\beta)$ is the only possible way to describe an E-pattern language by NE-pattern languages.

Theorem 5. *Let Σ be an alphabet with $|\Sigma| \geq 2$ and let α, β and γ be patterns. Then $L_{\mathrm{NE},\Sigma}(\alpha) \cup L_{\mathrm{NE},\Sigma}(\beta) = L_{\mathrm{E},\Sigma}(\gamma)$ if and only if $\alpha \in \Sigma^+$ and $\beta = \gamma = u_1\, x^{j_1}\, u_2\, x^{j_2} \ldots x^{j_m}\, u_{m+1}$, $j_i \in \mathbb{N}_0$, $1 \leq i \leq m$, such that $u_1\, u_2 \ldots u_{m+1} = \alpha$.*

With respect to the case $Z = \mathrm{E}$, $Z' = \mathrm{NE}$, we can even present a characterisation for the situation $L_{\mathrm{NE},\Sigma}(\alpha) = \bigcup_{i=1}^{k} L_{\mathrm{E},\Sigma}(\beta_i)$ with $k \leq |\Sigma|$. It shall be explained later on that this characterisation is a generalisation of the construction given by Jiang et al.

Theorem 6. *Let $\ell \geq 2$ and let Σ be an alphabet with $\{\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_\ell\} \subseteq \Sigma$. Furthermore, let $\alpha_1, \alpha_2, \ldots, \alpha_\ell$ and γ be patterns with $L_{\mathrm{E},\Sigma}(\alpha_i) \neq L_{\mathrm{E},\Sigma}(\alpha_j)$, $1 \leq i < j \leq \ell$. Then $\bigcup_{i=1}^{\ell} L_{\mathrm{E},\Sigma}(\alpha_i) = L_{\mathrm{NE},\Sigma}(\gamma)$ if and only if, for some permutation π of $(1, 2, \ldots, \ell)$,*

- *$\Sigma = \{\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_\ell\}$,*
- *$\gamma = u_1\, x\, u_2\, x\, u_3 \ldots u_k\, x\, u_{k+1}$, $k \geq 1$, $u_i \in \Sigma^*$, $1 \leq i \leq k+1$, and,*
- *for every i, $1 \leq i \leq \ell$,*

$$\alpha_i = u_1\, \alpha_i'\, \mathtt{a}_{\pi(i)}\, \alpha_i''\, u_2\, \alpha_i'\, \mathtt{a}_{\pi(i)}\, \alpha_i''\, u_3 \ldots u_k\, \alpha_i'\, \mathtt{a}_{\pi(i)}\, \alpha_i''\, u_{k+1}\,,$$

  where $\alpha_i', \alpha_i'' \in X^*$,
- *for every i, $1 \leq i \leq \ell$, there exists a $y_i \in \mathrm{var}(\alpha_i)$ with $|\alpha_i|_{y_i} = k$ and*
  - *$|\alpha_i'|_{y_i} = 1$ for all i, $1 \leq i \leq \ell$, or*
  - *$|\alpha_i''|_{y_i} = 1$ for all i, $1 \leq i \leq \ell$.*

If we apply the construction of Jiang et al. to a one-variable pattern γ, then we obtain patterns α_i, $1 \leq i \leq |\Sigma|$, that satisfy the conditions of the patterns in the statement of Theorem 6. More precisely, this corresponds to the special case where $\alpha_i'\alpha_i'' = y_i$, $1 \leq i \leq |\Sigma|$. Moreover, it can be easily verified that if γ and patterns α_i, $1 \leq i \leq |\Sigma|$, satisfy the conditions of the statement of Theorem 6, then, depending on whether $|\alpha_i'|_{y_i} = 1$ for all i, $1 \leq i \leq |\Sigma|$, or $|\alpha_i''|_{y_i} = 1$ for all i, $1 \leq i \leq |\Sigma|$, we can obtain patterns β_i from the patterns α_i by replacing $\alpha_i'\mathtt{a}_i\alpha_i''$ by $y_i\mathtt{a}_i$ or by $\mathtt{a}_i y_i$, respectively, and $\bigcup_{i=1}^{|\Sigma|} L_{\mathrm{E},\Sigma}(\beta_i) = L_{\mathrm{NE},\Sigma}(\gamma)$ still holds. Furthermore, the patterns β_i are exactly the patterns that are obtained if we apply the construction of Jiang et al.

Acknowledgments. The authors wish to thank the anonymous referees for their helpful suggestions, which have yielded a stronger version of Proposition 7.

References

1. Angluin, D.: Finding patterns common to a set of strings. Journal of Computer and System Sciences 21, 46–62 (1980)
2. Bayer, H.: Allgemeine Eigenschaften von Patternsprachen. Projektarbeit, Fachbereich Informatik, Universität Kaiserslautern (2007) (in German)
3. Fernau, H., Schmid, M.L.: Pattern matching with variables: A multivariate complexity analysis. In: Fischer, J., Sanders, P. (eds.) CPM 2013. LNCS, vol. 7922, pp. 83–94. Springer, Heidelberg (2013)
4. Fernau, H., Schmid, M.L., Villanger, Y.: On the parameterised complexity of string morphism problems. In: Proc. 33rd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2013. Leibniz International Proceedings in Informatics (LIPIcs), vol. 24, pp. 55–66 (2013)
5. Freydenberger, D., Reidenbach, D.: Bad news on decision problems for patterns. Information and Computation 208, 83–96 (2010)
6. Jain, S., Ong, Y., Stephan, F.: Regular patterns, regular languages and context-free languages. Information Processing Letters 110, 1114–1119 (2010)
7. Jiang, T., Kinber, E., Salomaa, A., Salomaa, K., Yu, S.: Pattern languages with and without erasing. International Journal of Computer Mathematics 50, 147–163 (1994)
8. Jiang, T., Salomaa, A., Salomaa, K., Yu, S.: Decision problems for patterns. Journal of Computer and System Sciences 50, 53–63 (1995)
9. Lange, S., Wiehagen, R.: Polynomial-time inference of arbitrary pattern languages. New Generation Computing 8, 361–370 (1991)
10. Reidenbach, D.: A non-learnable class of E-pattern languages. Theoretical Computer Science 350, 91–102 (2006)
11. Reidenbach, D.: An examination of Ohlebusch and Ukkonen's conjecture on the equivalence problem for E-pattern languages. Journal of Automata, Languages and Combinatorics 12, 407–426 (2007)
12. Reidenbach, D.: Discontinuities in pattern inference. Theoretical Computer Science 397, 166–193 (2008)
13. Reidenbach, D., Schmid, M.L.: Patterns with bounded treewidth. Information and Computation (to appear)
14. Reidenbach, D., Schmid, M.L.: Regular and context-free pattern languages over small alphabets. Theoretical Computer Science 518, 80–95 (2014)
15. Shinohara, T.: Inferring unions of two pattern languages. Bulletin of Informatics and Cybernetics 20, 83–88 (1983)
16. Shinohara, T., Arimura, H.: Inductive inference of unbounded unions of pattern languages from positive data. Theoretical Computer Science 241, 191–209 (2000)
17. Wright, K.: Identification of unions of languages drawn from an identifiable class. In: Proc. 2nd Annual Workshop on Computational Learning Theory, COLT 1989, pp. 328–333 (1989)