TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik

D. Kirsten[1]   **J. Marcinkowski**[2]

[1] Institut für Softwaretechnik I
Grundlagen der Programmierung
[2] University of Wrocław, Poland
Institute of Computer Science

Technische Berichte

Technical Reports

**Two Techniques in the Area
of the Star Problem**

Two Techniques in the Area
of the Star Problem

Daniel Kirsten[*]
Institute of Software Engineering I
Department of Computer Science
Dresden University of Technology
D-01062 Dresden, Germany

Daniel.Kirsten@inf.tu-dresden.de
http://www.inf.tu-dresden.de/~dk11

Jerzy Marcinkowski[†]
Institute of Computer Science
University of Wrocław
ul. Przesmyckiego 20
51-165 Wrocław, Poland

jma@tcs.uni.wroclaw.pl
http://www.tcs.uni.wroc.pl/~jma

December 31, 1999

Abstract

This paper deals with decision problems related to the star problem in trace monoids, which means to determine whether the iteration of a recognizable trace language is recognizable. Due to a theorem by G. RICHOMME from 1994 [32, 33], we know that the star problem is decidable in trace monoids which do not contain a submonoid of the form $\{a, c\}^* \times \{b, d\}^*$.

Here, we consider a more general problem: Is it decidable whether for some recognizable trace language $\mathbb{R}$ and some recognizable or finite trace language $\mathbb{P}$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable? If $\mathbb{P}$ is recognizable, then we show that this problem is decidable iff the underlying trace monoid does not contain a submonoid of the form $\{a, c\}^* \times b^*$. In the case of finite languages $\mathbb{P}$, we show several decidability and undecidability results.

Contents

1   Introduction

Free partially commutative monoids, also called trace monoids, were introduced by P. CARTIER and D. FOATA in 1969 [5]. In 1977, A. MAZURKIEWICZ proposed trace monoids as a potential model for concurrent processes [25], which marks the beginning of a systematic study of trace monoids by mathematicians and theoretical computer scientists, see e.g., [7, 8].

One main stream in trace theory is the study of recognizable trace languages, which can be considered as an extension of the well studied concept of regular languages in free monoids. A major step in this research is E. OCHMAŃSKI's PhD thesis from 1984 [30]. Some of the results concerning regular languages in free monoids can be generalized to recognizable languages in trace monoids. However, there is one major difference: The iteration of a recognizable trace language does not necessarily yield a recognizable language. This fact raises the so called star problem: Given a recognizable language L, is L^* recognizable? In general, it is not known whether the star problem is decidable. The main result after a stream of publications dealing with this problem is a theorem stated by G. RICHOMME in 1994, saying that the star problem is decidable in trace monoids which do not contain a particular submonoid called C4 [32, 33]. It is not known whether the star problem is decidable in trace monoids with a C4-submonoid. It is even unknown for finite trace languages.

Recently, D. KIRSTEN has shown that in trace monoids with a C4-submonoid it is undecidable whether for two recognizable trace languages $\mathbb{R}$ and $\mathbb{P}$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ yields a recognizable language [19, 22]. He also remarked that this problem is decidable in trace monoids without a P3-submonoid due to results by J. SAKAROVITCH. Here, we show that this problem is already undecidable in the trace monoids with a P3-submonoid. Consequently, the trace monoids without a P3-submonoid are exactly the trace monoids in which we can decide recognizability of $\mathbb{R} \cap \mathbb{P}^*$ for recognizable languages $\mathbb{R}$ and $\mathbb{P}$.

Secondly, we improve D. KIRSTEN's result to finite languages $\mathbb{P}$. In fact, in trace monoids with a C4-submonoid, we cannot decide recognizability of $\mathbb{R} \cap \mathbb{P}^*$ for recognizable languages $\mathbb{R}$ and *finite* languages $\mathbb{P}$.

Surprisingly, the combination of both improvements yields a decidable problem: In P3, i.e. in $\{a, b\}^* \times b^*$, we can decide for a recognizable language $\mathbb{R}$ and a *finite* language $\mathbb{P}$ whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable. Moreover, provided that $\mathbb{P}$ contains a trace of the form $\binom{\lambda}{b+}$, we can decide whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable.

The paper is organized as follows. After this introduction, Section 2 gives a formal overview consisting of an explanation of notions from algebra, formal language theory up to an overview of the star problem. In Part 2.5, we state our results in complete detail and discuss some open questions. In Section 3, we recall some classic results concerning automata and recognizable languages which we will use in our main proofs. In Sections 4 and 5, we give the proofs of the decidability and undecidability results, respectively.

2   Formal Definitions

2.1   Preliminaries

We introduce some notions from algebra and trace theory. By $\mathbb{N}$, we denote the set $\{0, 1, 2, \dots\}$. We allow to denote some singleton set by its element, e.g., we write 5 to denote both the number five and the singleton set consisting of the number five.

Assume some monoid $\mathbb{M}$. We denote its identity by $\lambda_{\mathbb{M}}$, or shortly by λ. Usually, we denote the product in $\mathbb{M}$ by juxtaposition but sometimes by $\cdot$ or $\cdot_{\mathbb{M}}$ to avoid confusion.

For every $n \in \mathbb{N}$ and $m \in \mathbb{M}$, we define the n-fold product by $m^0 = \lambda_{\mathbb{M}}$ and $m^{n+1} = m^n m$. We extend the product and the n-fold product to subsets of $\mathbb{M}$ as usual. Note that for every subset

$L \subseteq \mathbb{M}$, we have $L^0 := \{\lambda_{\mathbb{M}}\}$. In particular, $\emptyset^0 = \{\lambda_{\mathbb{M}}\}$. For subsets $L \subseteq \mathbb{M}$, we define the non-empty iteration L^+ as the union $L^1 \cup L^2 \cup L^3 \cup \ldots$ We denote the iteration of L by L^* and define it by $L^* := \lambda_{\mathbb{M}} \cup L^+$. For integers $i \leq j$, we denote by $L^{i,\ldots,j}$ the union $L^i \cup L^{i+1} \cup \ldots \cup L^j$.

Assume two monoids $\mathbb{M}$ and $\mathbb{M}'$. We denote their cartesian product by $\mathbb{M} \times \mathbb{M}'$. For some subsets $L \subseteq \mathbb{M}$ and $L' \subseteq \mathbb{M}'$, we denote their cartesian product by $L \times L'$, or sometimes, by $\binom{L}{L'}$ to visualize the componentwise concatenation.

Again, assume two monoids $\mathbb{M}$ and $\mathbb{M}'$. We call a function $h : \mathbb{M} \rightarrow \mathbb{M}'$ a *homomorphism* iff h preserves the product and the identity. We extend the notion of homomorphisms to subsets of $\mathbb{M}$ as usual. We denote the *inverse* of some homomorphism h by h^{-1}. We call h an *isomorphism* iff h is both injective and surjective, i.e., iff for every $m \in \mathbb{M}'$, the set $h^{-1}(m)$ is a singleton. Then, we can regard h^{-1} as a homomorphism from $\mathbb{M}'$ to $\mathbb{M}$. We call two monoids $\mathbb{M}$ and $\mathbb{M}'$ *isomorphic* iff there exists an isomorphism between them.

2.2   Monoids, Languages and Traces

By an *alphabet*, we mean a finite set of symbols. We call its elements *letters*. Assume an alphabet Σ. We denote the free monoid over Σ by Σ^*. For every word $w \in \Sigma^*$, we call the number of letters of w the *length* of w, and denote it by $|w|$. We denote by $\mathrm{alph}(w)$ set of letters which occur in w.

P. CARTIER and D. FOATA introduced the concept of the free partially commutative monoids in 1969 [5]. In 1977, A. MAZURKIEWICZ considered this concept as a potential model for concurrent systems [25]. Since then, free partially commutative monoids are examined by both mathematicians and theoretical computer scientists. For a general overview, I recommend the surveys [7, 8].

We call a binary relation I over some alphabet Σ an *independence relation* iff I is irreflexive and symmetric. For every pair of letters a and b with aIb, we say that a and b are *independent*, otherwise a and b are *dependent*. We call the pair (Σ, I) an *independence alphabet*. We call two words $w_1, w_2 \in \Sigma^*$ equivalent w.r.t. I iff we can transform w_1 into w_2 by finitely many exchanges of independent adjacent letters which we denote by $w_1 \sim_I w_2$. For instance, if a and c are independent letters, *baacbac*, *bacabac*, and *bcaabca* are mutually equivalent words.

The relation $\sim_I$ is an congruence relation w.r.t. the concatenation. For every word $w \in \Sigma^*$, we denote by $[w]_I$ the congruence class of w. We call the factorization of the free monoid Σ^* under $\sim_I$ the *trace monoid* over Σ and I and denote it by $\mathbb{M}(\Sigma, I)$. We call its elements, i.e., the congruence classes $[w]_I$ *traces*, its subsets *trace languages* or shortly *languages*. The function $[ ]_I$ is a homomorphism from Σ^* to $\mathbb{M}(\Sigma, I)$. As long as no confusion arises, we omit the index I at $[ ]_I$. We denote by $[ ]_I^{-1}$ the inverse of the homomorphism $[ ]_I$, i.e, for any trace $t \in \mathbb{M}(\Sigma, I)$, $[t]_I^{-1}$ denotes the set of all words in the trace (congruence class w.r.t. $\sim_I$) t.

If I is the empty relation over Σ, then the trace monoid $\mathbb{M}(\Sigma, I)$ is isomorphic to the free monoid Σ^*. If I is the largest irreflexive relation over Σ, i.e., two letters a and b are independent iff a and b are different, then $\mathbb{M}(\Sigma, I)$ is isomorphic to the free commutative monoid over Σ.

Because the words in some trace differ only in the order of their letters, we can define the length $|t|$ and the alphabet $\mathrm{alph}(t)$ of some trace t as the length and the alphabet of any word in the trace (congruence class) t, respectively.

Assume some independence alphabet (Σ, I). Every subset $\Gamma \subseteq \Sigma$ induces some independence alphabet $\left(\Gamma \, , \, I \cap (\Gamma \times \Gamma) \right)$ which we denote for short by (Γ, I).

Assume that we can split Σ into two non-empty, disjoint subsets Γ and Δ such that $(\Gamma \times \Delta) \subseteq I$, i.e., we have aIb for any letters $a \in \Gamma$ and $b \in \Delta$. Then, we call the independence alphabet *non-connected*. The trace monoid $\mathbb{M}(\Sigma, I)$ is isomorphic to the cartesian product $\mathbb{M}(\Gamma, I) \times \mathbb{M}(\Delta, I)$.

Conversely, assume two independence alphabets (Γ, I_Γ) and (Δ, I_Δ) such that $\Gamma \cap \Delta = \emptyset$. The cartesian product $\mathbb{M}(\Gamma, I_\Gamma) \times \mathbb{M}(\Delta, I_\Delta)$ is isomorphic to the trace monoid $\mathbb{M}(\Sigma, I)$ where

$$\Sigma = \Gamma \cup \Delta \qquad \text{and} \qquad I = I_\Sigma \cup I_\Delta \cup (\Gamma \times \Delta) \cup (\Delta \times \Gamma).$$

Hence, we can regard the cartesian product of two (disjoint) trace monoids as a trace monoid. Later, two trace monoids will play a crucial role: the so-called P3 and C4 which are defined (up to isomorphism) by $\{a, c\}^* \times b^*$ and $\{a, c\}^* \times \{b, d\}^*$, respectively.

We call some independence alphabet (Σ, I) *connected* if we cannot split Σ into two non-empty, disjoint subsets Γ and Δ with $(\Gamma \times \Delta) \subseteq I$.

For some independence alphabet (Σ, I), we call some trace $t \in \mathbb{M}(\Sigma, I)$ *non-connected* (resp. *connected*) if the independence alphabet $(\mathrm{alph}(t), I)$ is non-connected (resp. connected). Equivalently, some trace $t \in \mathbb{M}(\Sigma, I)$ is non-connected iff there are two non-empty traces $t_1, t_2 \in \mathbb{M}(\Sigma, I)$ with $t = t_1 t_2$ such that $\mathrm{alph}(t_1) \times \mathrm{alph}(t_2) \subseteq I$. Some trace $\binom{u}{v}$ in P3 or C4 is connected iff u or v is the empty word λ. For some trace language $L \in \mathbb{M}(\Sigma, I)$, we denote by $\mathrm{NC}(L)$ and $\mathrm{Cn}(L)$ the set of the connected and non-connected traces in L, respectively. We call some trace language L *connected* iff $\mathrm{NC}(L) = \emptyset$.

We call a homomorphism h between two trace monoids $\mathbb{M}(\Sigma, I_\Sigma)$ and $\mathbb{M}(\Delta, I_\Delta)$ *connected* iff for every $t \in \mathrm{Cn}(\mathbb{M}(\Sigma, I_\Sigma))$, we have $h(t) \in \mathrm{Cn}(\mathbb{M}(\Delta, I_\Delta))$

Assume traces $t_1, t_2 \in \mathbb{M}(\Sigma, I)$. We call t_1 a *prefix* of t_2 (for short $t_1 \sqsubseteq t_2$) iff $t_2 \in t_1 \mathbb{M}(\Sigma, I)$, i.e., iff there is some trace $s \in \mathbb{M}(\Sigma, I)$ such that $t_2 = t_1 s$. We call t_1 a *proper prefix* of t_2 (for short $t_1 \sqsubset t_2$) iff $t_1 \sqsubseteq t_2$ and $t_1 \neq t_2$. We call t_1 and t_2 *prefix-consistent* (for short $t_1 \sqcap t_2$) iff there is some $t \in \mathbb{M}(\Sigma, I)$ with $t_1 \sqsubseteq t$ and $t_2 \sqsubseteq t$, i.e., iff the languages $t_1 \mathbb{M}(\Sigma, I)$ and $t_2 \mathbb{M}(\Sigma, I)$ are not disjoint. If $\mathbb{M}(\Sigma, I)$ is a free monoid, then $t_1 \sqcap t_2$ iff $t_1 \sqsubseteq t_2$ or $t_2 \sqsubseteq t_1$.

Assume that $\mathbb{M}(\Sigma, I)$ is isomorphic to some cartesian product $\mathbb{M}(\Delta, I) \times \mathbb{M}(\Gamma, I)$. Then, two traces $\binom{u_1}{v_1}, \binom{u_2}{v_2} \in \mathbb{M}(\Sigma, I)$ are prefix-consistent iff $u_1 \sqcap u_2$ and $v_1 \sqcap v_2$.

Assume two alphabets Δ and Γ and a homomorphism $h : \Delta^* \to \Gamma^*$. We call h a *prefix-homomorphism* iff for any letters $a \neq b$ in Δ we have $h(a) \not\sqsubseteq h(b)$. We call h a *biprefix-homomorphism* iff additionally for any letters $a \neq b$ in Δ there is not any $w \in \Gamma^*$ with $wh(a) = h(b)$.

2.3 Automata and Recognizable Sets

We introduce the notion of recognizable languages as far as we need it in this paper, for a more general overview we recommend [2, 10].

Assume some monoid $\mathbb{M}$. An $\mathbb{M}$-*automaton* is a triple $\mathcal{A} = [Q, h, F]$, where Q is a finite monoid, h is a homomorphism $h : \mathbb{M} \to Q$ and F is a subset of Q. We define its language by $L(\mathcal{A}) = h^{-1}(F)$. We call Q the monoid of $\mathcal{A}$ and the elements of Q *states*. We call F the set of accepting states of $\mathcal{A}$ and h the homomorphism of $\mathcal{A}$. Without loss of generality, we can assume that h is a surjective homomorphism from $\mathbb{M}$ to Q.

We call some subset (resp. language) $L \subseteq \mathbb{M}$ *recognizable* iff there is some $\mathbb{M}$-automaton with $L = L(\mathcal{A})$. We denote the class of all recognizable sets over $\mathbb{M}$ by $\mathrm{REC}(\mathbb{M})$. In free monoids, recognizable languages are usually called *regular* languages.

It is a classic result that for any monoid $\mathbb{M}$, $\mathrm{REC}(\mathbb{M})$ contains the empty set $\emptyset$, $\mathbb{M}$ itself and it is closed under union, intersection, complement, and inverse homomorphisms [2, 10]. We need a theorem by J. Mezei concerning recognizable sets in Cartesian Products, cf. [2, 10].

Theorem 2.1 Assume two monoids $\mathbb{M}$ and $\mathbb{M}'$. Some set T is recognizable in $\mathbb{M} \times \mathbb{M}'$ iff there are an integer n, sets $K_1, \ldots, K_n \in \mathrm{REC}(\mathbb{M})$ and sets $L_1, \ldots, L_n \in \mathrm{REC}(\mathbb{M}')$ such that we have $T = (K_1 \times L_1) \cup \ldots \cup (K_n \times L_n)$. $\square$

The next lemma shows a widely used technique (cf. [2]).

Lemma 2.2 Assume some monoid $\mathbb{M}$ and recognizable sets $L_1, \ldots, L_n \subseteq \mathbb{M}$ for some $n > 0$. There are a finite monoid Q, a surjective homomorphism $h : \mathbb{M} \to Q$, and sets $F_1, \ldots, F_n \subseteq Q$ such that for $i \in \{1, \ldots, n\}$, the automaton $[Q, h, F_i]$ defines L_i. $\square$

Proof: Assume for $i \in \{1, \ldots, n\}$ automata $[Q_i, h_i, G_i]$ for L_i. We define $Q = Q_1 \times \ldots \times Q_n$. The homomorphism h maps every $p \in \mathbb{M}$ to $(h_1(p), \ldots, h_n(p))$. For $i \in \{1, \ldots, n\}$ we define $F_i = Q_1 \times \ldots \times Q_{i-1} \times G_i \times Q_{i+1} \times \ldots \times Q_n$. It is an easy verification that for $i \in \{1, \ldots, n\}$ the automaton $[Q, h, F_i]$ defines L_i. To obtain a surjective homomorphism h, we have to restrict Q and F_i for $i \in \{1, \ldots, n\}$ to $h(\mathbb{M})$ and $h(\mathbb{M}) \cap F_i$, respectively.   □

For any trace monoid $\mathbb{M}(\Sigma, I)$, $\mathrm{REC}(\mathbb{M}(\Sigma, I))$ contains all finite subsets of $\mathbb{M}(\Sigma, I)$ and is closed under concatenation [11] and iteration of *connected* recognizable trace languages [30, 6, 26]. In trace monoids, recognizable languages are not closed under homomorphisms. However, we have the following theorem:

Theorem 2.3 Assume two trace monoids $\mathbb{M}(\Sigma, I_\Sigma)$ and $\mathbb{M}(\Delta, I_\Delta)$, a homomorphism $h : \mathbb{M}(\Sigma, I_\Sigma) \to \mathbb{M}(\Delta, I_\Delta)$, and some language $L \subseteq \mathbb{M}(\Sigma, I_\Sigma)$.

1. If h is connected, then recognizability of L implies recognizability of $h(L)$.

2. If h is injective, then recognizability of $h(L)$ implies recognizability of L.   □

Assertion (1) is due to C. DUBOC [9]. It is a generalization of the well-known fact that homomorphisms between free monoids preserve recognizability. Assertion (2) is obvious, because we have $L = h^{-1}(h(L))$ and the closureship of recognizable sets under inverse homomorphisms. The survey article [29] gives an overview on recognizable trace languages including proofs of the closure properties.

We need the following useful lemma concerning recognizable trace languages and the notions of connected and non-connected traces.

Lemma 2.4 Assume some trace monoid $\mathbb{M}(\Sigma, I)$. Some language $L \subseteq \mathbb{M}(\Sigma, I)$ is recognizable iff both $\mathrm{NC}(L)$ and $\mathrm{Cn}(L)$ are recognizable.   □

Proof: We have $L = \mathrm{NC}(L) \cup \mathrm{Cn}(L)$, i.e., recognizability of both $\mathrm{NC}(L)$ and $\mathrm{Cn}(L)$ implies recognizability of L.

Assume that L is recognizable. We have $\mathrm{NC}(L) = L \cap \mathrm{NC}(\mathbb{M}(\Sigma, I))$ and $\mathrm{Cn}(L) = L \setminus \mathrm{NC}(L)$, i.e., it suffices to show that $\mathrm{NC}(\mathbb{M}(\Sigma, I))$ is recognizable. We can construct an $\mathbb{M}(\Sigma, I)$-automaton $[2^\Sigma, \mathrm{alph}, F]$ for $\mathrm{NC}(\mathbb{M}(\Sigma, I))$, where 2^Σ denotes the power set of Σ with set union as product. The set F consists of the subsets $\Gamma \subseteq \Sigma$ such that induced subalphabet (Γ, I) is non-connected. □

Let us shortly mention the notion of rational sets. Assume some monoid $\mathbb{M}$. The set of rational expressions $\mathrm{REX}(\mathbb{M})$ is the smallest set which contains the symbol $\emptyset$, the elements in $\mathbb{M}$ and is closed as follows: For some expressions $r, r_1, r_2 \in \mathrm{REX}(\mathbb{M})$, the expressions r^*, $(r_1 \cup r_2)$, and $(r_1 r_2)$ belong to $\mathrm{REX}(\mathbb{M})$. Every rational expression r defines a language $L(r)$ as usual.

We have KLEENE's classic result which asserts that in free monoids the recognizable sets and the rational sets coincide [37]. In trace monoids, we have just one direction due to a more general result by J. MCKNIGHT [2, 10]: Every recognizable trace language is rational. Moreover, we can transform every automaton into a rational expression which defines the same language. However, there are rational trace languages which are not recognizable unless the underlying trace monoid is a free monoid. See [3] for more information on rational trace languages.

2.4 The Star Problem

The following questions concerning the gap between the classes of recognizable and rational languages in trace monoids arise:

Recognizability Problem: Can we decide whether the language of a rational expression is a recognizable language?

Star Problem: Can we decide whether the iteration of a recognizable language yields a recognizable language?

J. SAKAROVITCH answered the first question in 1992.

Theorem 2.5 Assume a trace monoid $\mathbb{M}(\Sigma, I)$. The following three assertions are equivalent:

(1) $\mathbb{M}(\Sigma, I)$ does not contain an P3-submonoid.

(2) The rational languages of $\mathbb{M}(\Sigma, I)$ form an (effective) Boolean algebra.

(3) We can decide whether the language of a rational expression yields a recognizable language. $\square$

The equivalence of (1) and (2) is proved in [4, 1, 35], assertion (3) is added in [36].

During the recent 16 years, many papers have dealt with the star problem. However, only partial results have been achieved. We give a brief survey about its history. The star problem in the free monoid is trivial due to S. C. KLEENE, and it is decidable in free commutative monoids due to S. GINSBURG and E. SPANIER [15, 16]. In 1984, E. OCHMAŃSKI examined recognizable trace languages in his PhD thesis [30] and stated the star problem. During the eighties, E. OCHMAŃSKI [30], M. CLERBOUT and M. LATTEUX, [6] and Y. MÉTIVIER [26] independently proved that the iteration of a connected recognizable trace language yields a recognizable trace language. In 1992, J. SAKAROVITCH found the solution of the recognizability problem shown in Theorem 2.5. This solution implies the decidability of the star problem in trace monoids which do not contain a P3-submonoid. The attempt to extend SAKAROVITCH's characterization to the star problem failed, just in the same year, P. GASTIN, E. OCHMAŃSKI, A. PETIT, and B. ROZOY showed the decidability of the star problem in P3 [12].

During the subsequent years, Y. MÉTIVIER and G. RICHOMME developed these ideas. They showed decidability of the star problem for trace languages containing at most four traces as well as for finite sets containing at most two connected traces [27, 28]. Finally, G. RICHOMME proved the following theorem [24, 33].

Theorem 2.6 The star problem is decidable in trace monoids without C4-submonoid. $\square$

Recently, D. KIRSTEN and G. RICHOMME showed the equivalence between the star problem and the so-called finite power problem, which means to determine whether some recognizable language L has the finite power property, i.e., whether there is some integer n such that $L^* = L^0 \cup L^1 \cup \ldots \cup L^n$. Further, they reduced the remaining cases of the star problem to some particular languages in certain trace monoids [24].

Recently, D. KIRSTEN introduced the so-called generalized star problem (for short GSP) [19, 22]:

Generalized Star Problem: Can we decide whether for two recognizable languages $\mathbb{R}$, $\mathbb{P}$ in some trace monoid the intersection $\mathbb{R} \cap \mathbb{P}^*$ yields a recognizable language?

In the particular case that $\mathbb{R}$ is the complete trace monoid, the GSP is exactly the star problem. As a conclusion from Theorem 2.5, the GSP is decidable in trace monoids without P3-submonoid. To decide whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable, we construct rational expressions for $\mathbb{R}$ and $\mathbb{P}$, resp., a rational expression for $\mathbb{R} \cap \mathbb{P}^*$ by Theorem 2.5(2) and finally, we determine by Theorem 2.5(3) whether this expression defines a recognizable language.

Recently, D. KIRSTEN has shown the following result [19, 22]:

Theorem 2.7 Assume some trace monoid $\mathbb{M}(\Sigma, I)$ with a C4-submonoid. The GSP is undecidable in $\mathbb{M}(\Sigma, I)$. $\square$

2.5 Main Results, Conclusions, and Future Steps

In this paper, we show various improvements of Theorem 2.7. We also show some decidability results for particular cases of the GSP. Weaker versions of our theorems were already announced in [23]. Theorem 2.10 already occurred in [21] among other results.

On one hand, we show the following improvements of Theorem 2.7:

Theorem 2.8

1. It is undecidable whether for some recognizable language $\mathbb{P} \subseteq P3 = \{a, c\}^* \times \{b\}^*$ the intersection $\binom{a\{a,c\}^*}{b^*} \cap \binom{\{a,c\}^*a}{b^*} \cap \mathbb{P}^*$ is recognizable.

2. There is some fixed recognizable language $\mathbb{R} \subseteq C4$ such that it is undecidable whether for some finite language $\mathbb{P} \subseteq C4$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable.

3. It is undecidable whether for some recognizable language $\mathbb{P} \subseteq C4 = \{a, c\}^* \times \{b, d\}^*$ the intersection $\binom{a\{a,c\}^*}{\{b,d\}^*} \cap \mathbb{P}^*$ is recognizable.   □

Note that the intersection in Assertion (1) just means to select the traces from $\mathbb{P}^*$ whose first component starts and ends with the letter a. We prove Assertion (1) in Part 5.1 by a reduction to some undecidable problem concerning picture languages. Because the GSP is decidable in trace monoids without P3-submonoid, we obtain the following corollary from Assertion (1):

Corollary 2.9 Assume some trace monoid $\mathbb{M}(\Sigma, I)$. It is decidable whether for two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq \mathbb{M}(\Sigma, I)$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable *iff* $\mathbb{M}(\Sigma, I)$ does not contain a P3-submonoid.   □

In Part 5.2, we show Assertions (2) and (3) by a reduction to a variant of the PCP. An example for some language $\mathbb{R}$ in Assertion (2) is the language

$$\mathbb{R} = \begin{pmatrix} \{aaacaacccacca, aaacaacccaccc\}^+ aa \\ \{bbbdbbddbdb, bbbdbbdddbddd\}^+ bb \end{pmatrix}$$

(cf. Remark 5.13 on page 23.) On the other hand, at least in the trace monoid P3, the GSP is decidable as far as $\mathbb{P}$ satisfies some property:

Theorem 2.10 It is decidable whether for two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq P3 = \{a, c\}^* \times \{b\}^*$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable, provided that $\mathbb{P}$ satisfies *at least one* of the following properties:

1. There is some integer n such that $NC(\mathbb{P}) \subseteq \binom{\{a,c\}^*}{b^{1,\ldots,n}}$,   or

2. some trace of the form $\binom{\lambda}{b^+}$ belongs to $\mathbb{P}$.   □

We give its proof in Section 4. We will use HASHIGUCHI's distance automata in a crucial way as well as some pumping techniques. For lucidity, we state the following corollary which is an obvious conclusion from Theorem 2.10:

Corollary 2.11 It is decidable whether for two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq P3 = \{a, c\}^* \times \{b\}^*$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable, provided that $NC(\mathbb{P})$ is finite.   □

Clearly, if $NC(\mathbb{P})$ is finite, then $\mathbb{P}$ satisfies Property (1) in Theorem 2.10. However, $\mathbb{P} = \binom{\{a,c\}^*}{b}$ satisfies Property (1) in Theorem 2.10, although $NC(\mathbb{P})$ is infinite, i.e., Corollary 2.11 is weaker than Theorem 2.10.

Although Theorem 2.8(3) seems to be close to an answer to the star problem, we do not know whether it is undecidable in C4. We regard the star problem as the most important open

question in this area. We leave it to the reader to use our results to obtain a conjecture for the star problem. Corollary 2.9 states a characterization of the trace monoids with a decidable GSP. An open question is to characterize the trace monoids in which the GSP is decidable under the restriction to finite languages $\mathbb{P}$. For instance, in the trace monoid $\{a, c\}^* \times c^* \times d^* = P3 \times d^*$ we can neither apply Theorem 2.8(2) nor Corollary 2.11. The attempt to show some variants of Theorem 2.10 and Corollary 2.11 for $P3 \times d^*$ by adapting the proof ideas for Theorem 2.10 leads to serious problems, e.g., to some notion of HASHIGUCHI's distance automata over $P3$ ...

One more remaining problem is to show some common improvement of Theorem 2.8(2) and Theorem 2.8(3), i.e., to show Theorem 2.8(2) for $\mathbb{R} = \binom{a\{a,c\}^*}{\{b,d\}^*}$ which means to show Theorem 2.8(3) for finite languages $\mathbb{P}$.

Finally, a question is whether one can show Theorem 2.8(1) for the intersection $\binom{a\{a,c\}^*}{b^*} \cap \mathbb{P}^*$.

3 Some Classic Results

In this section, we recall some classic notions and results which we will use in our main proofs. In Part 3.1, we deal with transition automata by M. O. RABIN and D. SCOTT and distance automata by K. HASHIGUCHI. Then, we consider transition automata over trace monoids.

In Part 3.3, we recall picture languages and finally, in Part 3.4, we deal with a variant of POST's Correspondence Problem which will be very suitable in the proof of Theorem 2.8.

3.1 Transition Automata over Free Monoids

We recall some notions from automata theory. At first, we deal with two technical lemmas concerning recognizable languages in free monoids.

Lemma 3.1 Assume some alphabet Σ and some automaton $[Q, h, F]$ such that $h : \Sigma^* \to Q$ is a surjection. For every $q \in Q$, there is some word $w \in h^{-1}(q)$ with $|w| < |Q|$. $\qquad\square$

Proof: Assume some $q \in Q$ and some $w \in h^{-1}(q)$. If $|w| < |Q|$, then we are done. Assume $|w| \geq |Q|$. Choose letters $a_1, \ldots, a_{|w|} \in \Sigma$ such that $a_1 \ldots a_{|w|} = w$. Because $|w| \geq |Q|$, there are two integers $0 \leq i < j \leq |w|$ with $h(a_1 \ldots a_i) = h(a_1 \ldots a_j)$, i.e., $q = h(a_1 \ldots a_{|w|}) = h(a_1 \ldots a_i a_{j+1} \ldots a_{|w|})$ and $|a_1 \ldots a_i a_{j+1} \ldots a_{|w|}| < |w|$. By applying such a cut as many times a necessary, we construct some word w' with $|w'| < |Q|$ and $h(w') = q$. $\qquad\square$

See [10, p. 101] for a proof of the following lemma.

Lemma 3.2 Some language $L \subseteq b^*$ is recognizable iff there are some finite language $L_0 \subseteq b^*$ and integers z, n, $m_1, \ldots, m_n$ such that $L = \bigcup_{i \in \{0,\ldots,n\}} L_i$ with $L_i = b^{m_i}(b^z)^*$ for $i \in \{1, \ldots, n\}$. $\qquad\square$

Clearly, we can assume $z, m_1, \ldots, m_n > 0$. If L is given by some automaton, then we can construct L_0 and the integers z, n, and $m_1, \ldots, m_n$.

Transition automata originate from M. O. RABIN and D. SCOTT [31]. See [2, 10] for more information. A *transition automaton* is a quadruple $\mathcal{A} = [Q, s, E, F]$, where

- Q is a finite set called the *states*,

- $s \in Q$ is called the *initial state*,

- $E \subseteq Q \times \Sigma \times Q$ is a set called the *edges*, and

- $F \subseteq Q$ are called the *accepting states*.

We call a *path* in $\mathcal{A}$ a finite sequence of edges $(q_1, a_1, q_2)(q_2, a_2, q_3) \ldots (q_n, a_n, q_{n+1})$ for some $n \geq 0$. We call the word $a_1 \ldots a_n$ the *label* of this path. We call a path *accepting* iff $q_1 = s$ and $q_{n+1} \in F$. The *language of* $\mathcal{A}$, denoted by $L(\mathcal{A})$, consists of the labels of accepting paths.

Sometimes, it is quite convenient to consider transition automata as devices which process with some (read-only) head over a tape. In the beginning, the automaton rests in the initial state s, the tape contains some word $w \in \Sigma^*$, and the head of the automaton is over the first letter of w. If the first letter of w is a and there is some edge $(s, a, q) \in E$, then the automaton can read a, i.e., it can change the state to q and move the head to the second letter of w.

It is a classic result in automata theory that transition automata over free monoids define exactly the recognizable languages (cf. [2, 10]). Moreover, we can transform every transition automaton into a Σ^*-automaton which defines the same language, and vice versa. Further, for every recognizable language $L \subseteq \Sigma^*$ with $\lambda \notin L$, we can construct a transition automaton $[Q, s, E, F]$ for L such that

- $|F| = 1$ and

- $E \subseteq (Q \setminus F) \times \Sigma \times (Q \setminus s)$

provided that L is given by, e.g., some transition automaton or some Σ^*-automaton (cf. [2, 10]). We can generalize transition automata by allowing that E is a finite subset of $Q \times \Sigma^+ \times Q$. Then, the label of some path $(q_1, u_1, q_2)(q_2, u_2, q_3) \ldots (q_n, u_n, q_{n+1})$ is the concatenation $u_1 \ldots u_n$. As above, we define the language of such an automaton as the set of the labels of any accepting path. We can transform some transition automaton $[Q, s, E, F]$ with $E \subseteq (Q \times \Sigma^+ \times Q)$ into a transition automaton $[Q', s, E', F]$ with $E' \subseteq (Q' \times \Sigma \times Q')$ which defines the same language. If we consider such an automaton as a device over some tape, then this generalization simply means that the automaton can read several letters in one step.

We can further generalize transition automata by allowing that E is some (not necessarily finite) subset of $Q \times \Sigma^+ \times Q$. However, these generalized transition automata exceed the concept of recognizable languages in Σ^*. Nevertheless, we will use them as a convenient tool in the proof of Proposition 4.4.

Distance automata were introduced by K. HASHIGUCHI [17, 18]. We assume some element ∞ which is bigger than every integer. A *distance automaton* is a tuple $\mathcal{A} = [Q, s, E, F, \delta]$ where

- $[Q, s, E, F]$ is a transition automaton, and

- $\delta : Q \times \Sigma \times Q \to \{0, 1, \infty\}$ is a function called *distance function* such that for every $q, q' \in Q$ and $a \in \Sigma$ we have $\delta(q, a, q') = \infty$ iff $(q, a, q') \notin E$.

Opposed to K. HASHIGUCHI who considered distance automata with several initial states, we just deal with distance automata with exactly one initial state.

We define the language of some distance automaton $\mathcal{A} = [Q, s, E, F, \delta]$ as the language of the transition automaton $[Q, s, E, F]$. We define the *distance* of some path $(q_1, a_1, q_2) \ldots (q_n, a_n, q_{n+1})$ as the sum $\delta(q_1, a_1, q_2) + \delta(q_2, a_2, q_3) + \ldots + \delta(q_n, a_n, q_{n+1})$ where the sign "+" denotes common integer addition. We denote the distance of some word $w \in L(\mathcal{A})$ by $\delta(w)$ and define it as the least integer n such that there is an accepting path in $\mathcal{A}$ with the label w and the distance n. We define the distance of every word $w \notin L(\mathcal{A})$ by $\delta(w) = \infty$. We call some distance automaton $\mathcal{A}$ *limited in distance* iff there is some integer $\delta_{\max}$ such that for every word $w \in L(\mathcal{A})$ we have $\delta(w) \leq \delta_{\max}$. We use the following strong result by K. HASHIGUCHI [17, 18] in a crucial way.

Theorem 3.3 It is decidable whether some distance automaton is limited in distance. $\square$

3.2 Transition Automata over Trace Monoids

We assume some trace monoid $\mathbb{M}(\Sigma, I)$ within this part. Transition automata over $\mathbb{M}(\Sigma, I)$ are defined as transition automata over Σ^*. However, the label of some path $(q_1, a_1, q_2)(q_2, a_2, q_3) \ldots (q_n, a_n, q_{n+1})$ is the trace $[a_1 a_2 \ldots a_n] \in \mathbb{M}(\Sigma, I)$. Transition automata over $\mathbb{M}(\Sigma, I)$ define exactly the rational languages over $\mathbb{M}(\Sigma, I)$.

A transition automaton $\mathcal{A} = [Q, s, E, F]$ respects I iff for every $p, q, r \in Q$ and for every independent $a, b \in \Sigma$ with $(p, a, q), (q, b, r) \in E$ we have some $q' \in Q$ and $(p, b, q'), (q', a, r) \in E$. Transition automata over $\mathbb{M}(\Sigma, I)$ which respect I define the recognizable languages over $\mathbb{M}(\Sigma, I)$.

Similarly to transition automata over free monoids, we can also allow that E is some finite or even infinite subset of $Q \times \mathbb{M}(\Sigma, I) \times Q$. However, infiniteness of E exceeds the concept of rational trace languages.

If $\mathbb{M}(\Sigma, I)$ is a isomorphic to a monoid $\Gamma^* \times \Delta^*$, then we can consider transition automata over $\Gamma^* \times \Delta^*$ as devices which process with two heads over two tapes, respectively. In the beginning, the automaton is in the initial state s, and some trace $\binom{u}{v} \in \Gamma^* \times \Delta^*$ is represented on the tapes, i.e., u and v are represented on the first and second tape, respectively. If there is some instruction $(s, a, q) \in E$, and u starts with some letter $a \in \Gamma$, then the automaton can read a, i.e., it changes its state to q, moves the first head to the second letter of u, and does not move the second head. If we allow instructions $Q \times (\Gamma^* \times \Delta^*) \times Q$, then the automaton can move both heads in one step.

3.3   Picture Languages

Pictures and picture languages are a generalization of words and word languages. We use some very basic notions from the theory of picture languages. See [14] for a recent survey. Assume some alphabet Γ. For two integers $m, n \geq 1$, a *picture p over* Γ of the size (m, n) is a mapping $p : \{1, \ldots, m\} \times \{1, \ldots, n\} \to \Gamma$. We call the numbers m and n the *height* and *width* of p and denote them by $\mathrm{hgt}(p)$ and $\mathrm{wdt}(p)$, respectively. For convenience, we define the mapping $\hat{p} : \{0, \ldots, m+1\} \times \{0, \ldots, n+1\} \to \Gamma \cup \{\#\}$. For every $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$, $\hat{p}_{i,j}$ yields $p_{i,j}$. Otherwise, $\hat{p}$ yields $\#$. We denote the set of all pictures over Γ by Γ^{**}.

Assume two pictures p and s. The *column concatenation* $p \circ s$ is defined iff $\mathrm{hgt}(p) = \mathrm{hgt}(s)$. Let us denote the size of p and s by (m, n) and (m, n'), respectively. Then, $p \circ s$ is defined by:

$$p \circ s = \begin{pmatrix} p_{1,1} & \cdots & p_{1,n} \; s_{1,1} & \cdots & s_{1,n'} \\ \vdots & \ddots & \vdots \quad \vdots & \ddots & \vdots \\ p_{m,1} & \cdots & p_{m,n} \; s_{m,1} & \cdots & s_{m,n'} \end{pmatrix}$$

The extension of the column concatenation to picture languages is obvious.

We call a set of pictures of the size $(2, 2)$ over $\Gamma \cup \{\#\}$ a *local representation* over Γ. Such a local representation defines a picture language $L(\Delta)$. A picture p belongs to $L(\Delta)$ iff every $(2, 2)$ sub-picture of p belongs to Δ:

$$L(\Delta) = \left\{ p \in \Gamma^{**} \;\middle|\; i \in \{0, \ldots, \mathrm{hgt}(p)\},\, j \in \{0, \ldots, \mathrm{wdt}(p)\} : \begin{pmatrix} \hat{p}_{i,j} & \hat{p}_{i,j+1} \\ \hat{p}_{i+1,j} & \hat{p}_{i+1,j+1} \end{pmatrix} \in \Delta \right\}.$$

We use the following theorem from [13].

Theorem 3.4 It is not decidable whether the language of a local representation is empty or whether it is finite. □

D. GIAMMARRESI and A. RESTIVO showed several closure properties of the class of picture languages of local representations [13]. We just need the following result:

Lemma 3.5 Assume an alphabet Γ and a letter $b \notin \Gamma$. We can transform some local representation Δ over Γ into a local representation Δ' over $\Gamma \cup \{b\}$ such that $L(\Delta') = L(\Delta) \circ \{b\}^{**}$. □

Proof (sketch): For any $a, c \in \Gamma$ we replace tiles $\left(\begin{smallmatrix} a & \# \\ c & \# \end{smallmatrix}\right), \left(\begin{smallmatrix} \# & \# \\ c & \# \end{smallmatrix}\right), \left(\begin{smallmatrix} a & \# \\ \# & \# \end{smallmatrix}\right) \in \Delta$ by $\left(\begin{smallmatrix} a & b \\ c & b \end{smallmatrix}\right), \left(\begin{smallmatrix} \# & \# \\ c & b \end{smallmatrix}\right), \left(\begin{smallmatrix} a & b \\ \# & \# \end{smallmatrix}\right)$, respectively, and we insert new tiles $\left(\begin{smallmatrix} \# & \# \\ b & b \end{smallmatrix}\right), \left(\begin{smallmatrix} b & b \\ b & b \end{smallmatrix}\right), \left(\begin{smallmatrix} b & b \\ \# & \# \end{smallmatrix}\right), \left(\begin{smallmatrix} \# & \# \\ b & \# \end{smallmatrix}\right), \left(\begin{smallmatrix} b & \# \\ b & \# \end{smallmatrix}\right), \left(\begin{smallmatrix} b & \# \\ \# & \# \end{smallmatrix}\right)$ into Δ'. □

3.4   A variant of POST's Correspondence Problem

POST's Correspondence Problem (for short PCP) is one of the most common undecidable problems. A *PCP instance* consists of two alphabets Δ and Σ and two homomorphisms $\alpha, \beta : \Delta^* \to \Sigma^*$. Assume such an instance for the rest of this part. A *solution* is a non-empty word $w \in \Delta^+$ such that $\alpha(w) = \beta(w)$. The existence of a solution is undecidable. An infinite sequence $i_1, i_2, i_3 \ldots$ of letters in Δ is called an *infinite solution* iff for any integer n, the words $\alpha(i_1 \ldots i_n)$ and $\beta(i_1 \ldots i_n)$ are prefix consistent. We have the following result due to K. RUOHONEN [34].

Theorem 3.6 Assume a PCP instance such that both α and β are biprefix homomorphisms. It is undecidable whether it has a solution and it is undecidable whether it has an infinite solution. □

We need the following lemma:

Lemma 3.7 A PCP instance has an infinite solution iff there are infinitely many words $w \in \Delta^*$ such that $\alpha(w)$ and $\beta(w)$ are prefix consistent. □

Proof: Assume a PCP-instance consisting of Δ, Σ, α, and β. If it has is an infinite solution $i_1, i_2, \ldots$, then we have $\alpha(i_1 \ldots i_n) \sqcap \beta(i_1 \ldots i_n)$ for $n \geq 0$, i.e., for infinitely many words $i_1 \ldots i_n$.

Conversely, let $L \subseteq \Delta^*$ denote the infinite language which consists of the words in $w \in \Delta^*$ such that $\alpha(w) \sqcap \beta(w)$. Clearly, L is prefix-closed. We inductively construct an infinite solution. Assume some integer n and some word $i_1 \ldots i_n \in \Delta^*$ such that $i_1 \ldots i_n$ satisfies two properties: Firstly, $\alpha(i_1 \ldots i_n) \sqcap \beta(i_1 \ldots i_n)$, i.e., $i_1 \ldots i_n \in L$. Secondly, $i_1 \ldots i_n$ is a prefix of infinitely many words in L. Then, there is at least one letter $i_{n+1} \in \Delta$ such that $i_1 \ldots i_{n+1}$ satisfies the same properties. We can use the empty word λ as initial value for the iteration. $\square$

4   Some Decidable Cases

In this section, we prove Theorem 2.10. Assume two disjoint alphabets Σ and Γ and some letter $b \notin \Sigma$ within this section. To prove Theorem 2.10, we have to show the decidability of a special case of the GSP, i.e., we have to show that the recognizability of $\mathbb{R} \cap \mathbb{P}^*$ is decidable for recognizable languages $\mathbb{R}, \mathbb{P} \subseteq \Sigma^* \times b^*$, provided that $\mathbb{P}$ is finite or some trace of the form $\binom{\lambda}{b^+}$ belongs to $\mathbb{P}$. Some of our intermediary results also hold for trace monoids of the form $\Sigma^* \times \Gamma^*$.

In Part 4.1, we consider some easy propositions which allow us to reduce the GSP to restricted languages $\mathbb{R}$. Then, we consider two cases of Theorem 2.10. In Part 4.2, we show the decidability of the GSP in trace monoids $\Sigma^* \times \Gamma^*$ restricted to recognizable languages $\mathbb{P} \subseteq (\Sigma^+ \times \Gamma^{0,\ldots,n})$ for some integer n. This includes the case that $\mathbb{P}$ is a finite subset of $\Sigma^+ \times \Gamma^*$.

In Part 4.3, we use HASHIGUCHI's distance automata to show the decidability of the GSP in trace monoids $\Sigma^* \times \Gamma^*$ provided that some trace of the form $\binom{\lambda}{b^+}$ belongs to $\mathbb{P}$.

4.1   Some Obvious Observations

For some finite language $\mathbb{R} \subseteq (\Sigma^* \times \Gamma^*)$ and any language $\mathbb{P} \subseteq (\Sigma^* \times \Gamma^*)$, the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable because it is finite. We generalize this obvious fact.

Proposition 4.1 Assume two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq (\Sigma^* \times \Gamma^*)$. The intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable if $\mathbb{R}$ satisfies one of the following conditions:

1. We have $\mathbb{R} \subseteq (\Sigma^* \times \Gamma^{0,\ldots,n})$ for some integer n, or

2. $\mathbb{R}$ is a connected language.   □

Proof: By Lemma 2.4, $\mathrm{NC}(\mathbb{P})$ and $\mathrm{Cn}(\mathbb{P})$ are recognizable. The concatenation of some traces $t_1, \ldots, t_m \in (\Sigma^* \times \Gamma^*)$ for some m yields a non-connected trace if one of the traces $t_1, \ldots, t_m$ is non-connected. Hence, if $\mathbb{R}$ is connected, then we have $\mathbb{R} \cap \mathbb{P}^* = \mathbb{R} \cap \mathrm{Cn}(\mathbb{P})^*$ which is recognizable by the closure properties of recognizable trace languages.

Assume that $\mathbb{R}$ satisfies (1). We have $\mathbb{P}^* = \mathrm{Cn}(\mathbb{P})^* \big(\mathrm{NC}(\mathbb{P})\mathrm{Cn}(\mathbb{P})^*\big)^*$. Because every non-connected trace in $\Sigma^* \times \Gamma^*$ contains at least one letter from Γ, we have

$$\mathbb{R} \cap \mathbb{P}^* = \mathbb{R} \cap \mathrm{Cn}(\mathbb{P})^* \big(\mathrm{NC}(\mathbb{P})\mathrm{Cn}(\mathbb{P})^*\big)^{0,\ldots,n}$$

This language is recognizable by the closure properties of recognizable trace languages.   □

Proposition 4.2 Assume two recognizable languages $\mathbb{R}, \mathbb{P}$ in some trace monoid. Assume some integer n and recognizable languages $\mathbb{R}_1, \ldots, \mathbb{R}_n$ with $\mathbb{R}_1 \cup \ldots \cup \mathbb{R}_n = \mathbb{R}$. Then, the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable iff for $i \in \{1, \ldots, n\}$ the intersection $\mathbb{R}_i \cap \mathbb{P}^*$ is recognizable.   □

Proof: For $i \in \{1, \ldots, n\}$, we have $\mathbb{R}_i \cap \mathbb{P}^* = \mathbb{R}_i \cap (\mathbb{R} \cap \mathbb{P}^*)$. Hence, recognizability of $\mathbb{R} \cap \mathbb{P}^*$ implies recognizability of $\mathbb{R}_i \cap \mathbb{P}^*$. Conversely, we have $\mathbb{R} \cap \mathbb{P}^* = (\mathbb{R}_1 \cap \mathbb{P}^*) \cup \ldots \cup (\mathbb{R}_n \cap \mathbb{P}^*)$. Thus, recognizability of $\mathbb{R}_i \cap \mathbb{P}^*$ for $i \in \{1, \ldots, n\}$ implies recognizability of $\mathbb{R} \cap \mathbb{P}^*$.   □

From these propositions, we immediately see that for two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq (\Sigma^* \times \Gamma^*)$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable, provided that $\mathrm{NC}(\mathbb{R})$ is finite.

4.2 A Decidable Case in $\Sigma^* \times \Gamma^*$

Now, we work on the GSP for restricted languages $\mathbb{P}$.

Proposition 4.3 Assume two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq (\Sigma^* \times \Gamma^*)$. We can decide whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable if $\mathbb{P} \subseteq (\Sigma^+ \times \Gamma^{0,\ldots,n})$ for some integer n. $\qquad\square$

We forbid that the empty trace belongs to $\mathbb{P}$. However, this is not really a restriction, because $\mathbb{P}^* = (\mathbb{P} \setminus \binom{\lambda}{\lambda}))^*$. Note that Proposition 4.3 includes the case that $\mathbb{P}$ is a finite subset of $\Sigma^+ \times \Gamma^*$.

Proof: Let $[Q, h, F]$ be some automaton for $\mathbb{R}$. We have $\mathrm{Cn}(\mathbb{P}) \subseteq (\Sigma^* \times \lambda)$. We abbreviate $\mathrm{Cn}(\mathbb{P})^* \mathrm{NC}(\mathbb{P}) \mathrm{Cn}(\mathbb{P})^*$ by $\mathbb{P}_{\mathrm{CNC}}$. We show the equivalence of three assertions:

1. The language $\mathbb{R} \cap \mathbb{P}^*$ is recognizable.

2. There is some integer n' such that $(\mathbb{R} \cap \mathbb{P}^*) \subseteq (\Sigma^* \times \Gamma^{0,\ldots,n'})$.

3. The intersection $\mathbb{R} \cap \mathbb{P}_{\mathrm{CNC}}^{|Q|+1,\ldots,2|Q|+1}$ is empty.

- (2)$\Rightarrow$(1) We have $\mathbb{R} \cap \mathbb{P}^* = (\Sigma^* \times \Gamma^{0,\ldots,n'}) \cap (\mathbb{R} \cap \mathbb{P}^*)$ which is $\left((\Sigma^* \times \Gamma^{0,\ldots,n'}) \cap \mathbb{R} \right) \cap \mathbb{P}^*$. By Proposition 4.1 (1), this language is recognizable.

- (1)$\Rightarrow$(2) Assume that $\mathbb{R} \cap \mathbb{P}^*$ is recognizable, but nevertheless, an integer n' in (2) does not exist. By MEZEI's Theorem, the intersection $\mathbb{R} \cap \mathbb{P}^*$ consists of finitely many cartesian products $(K \times L) \subseteq (\Sigma^* \times \Gamma^*)$ with $K \neq \emptyset$ and $L \neq \emptyset$. Because an integer n' in assertion (2) does not exist, we can choose a cartesian product $(K \times L) \subseteq (\mathbb{R} \cap \mathbb{P}^*)$ such that L is infinite. Choose some $w \in K$. We have $(w \times L) \subseteq (\mathbb{R} \cap \mathbb{P}^*) \subseteq \mathbb{P}^*$. Because every trace in $\mathbb{P}$ contains at least one letter in Σ, we have $(w \times L) \subseteq \mathbb{P}^{0,\ldots,|w|}$. Because every trace in $\mathbb{P}$ contains at most n occurrences of letters from Γ, the length of the words in L cannot exceed $n|w|$. This contradicts that L is infinite.

- (2)$\Rightarrow$(3) We assume that the intersection in (3) is not empty. Consequently, there is an integer $l \in \{ |Q|+1, \ldots, 2|Q|+1 \}$ and there are traces $t_1, \ldots, t_l \in \mathbb{P}_{\mathrm{CNC}} \subseteq \mathbb{P}^*$ such that $t_1 \ldots t_l \in \mathbb{R}$. Because $|Q| < l$, there are two integers i, j with $0 < i < j \leq l$ such that $h(t_1 \ldots t_i) = h(t_1 \ldots t_j)$. Then, "we can pump $h(t_{i+1} \ldots t_j)$". For $k \geq 0$, we have

$$h(t_1 \ldots t_i) = h(t_1 \ldots t_i)h(t_{i+1} \ldots t_j)^k \text{ and } h(t_1 \ldots t_l) = h(t_1 \ldots t_i)h(t_{i+1} \ldots t_j)^k h(t_{j+1} \ldots t_l)$$

This value belongs to F such that we have $(t_1 \ldots t_i)(t_{i+1} \ldots t_j)^*(t_{j+1} \ldots t_l) \subseteq \mathbb{R}$. We also have $(t_1 \ldots t_i)(t_{i+1} \ldots t_j)^*(t_{j+1} \ldots t_l) \subseteq \mathbb{P}^*$, because $t_1, \ldots, t_l \in \mathbb{P}^*$. The traces $t_{i+1}, \ldots, t_j$ contain at least one non-connected trace, i.e., they contain one letter from Γ. Hence, by pumping $t_{i+1} \ldots t_j$, we see that an integer n' as in assertion (2) cannot exist.

- (3)$\Rightarrow$(2) Let us assume that an integer n' does not exist. Every trace in $\mathbb{P}$ contains at most n occurrences of letters in Γ. However, there are traces in $\mathbb{R} \cap \mathbb{P}^*$ containing arbitrary many occurrences of letters in Γ. Consequently, there are arbitrary big integers l such that $\mathbb{P}_{\mathrm{CNC}}^l$ contains traces in $\mathbb{R}$. So assume an integer $l \geq |Q| + 1$ such that there are traces $t_1, \ldots, t_l \in \mathbb{P}_{\mathrm{CNC}}$ with $t_1 \ldots t_l \in \mathbb{R}$. If $l \leq 2|Q| + 1$, then we are done.

  So assume $l > 2|Q| + 1$. As above, there are two integers i, j with $0 < i < j \leq |Q| + 1$ such that $h(t_1 \ldots t_i) = h(t_1 \ldots t_j)$. We have $j - i \leq |Q|$. As above, we have $t_1 \ldots t_i t_{j+1} \ldots t_l \in \mathbb{R}$. Hence, $t_1 \ldots t_i t_{j+1} \ldots t_l$ belongs to the intersection $\mathbb{R} \cap \mathbb{P}_{\mathrm{CNC}}^{l-j+i}$. By applying such a cut as many times as necessary, we obtain some trace in $\mathbb{R} \cap \mathbb{P}_{\mathrm{CNC}}^{|Q|+1,\ldots,2|Q|+1}$.

Because the closure properties of recognizable trace languages are effective, we can construct an automaton for $\mathbb{R} \cap \mathbb{P}_{\mathrm{CNC}}^{|Q|+1,\ldots,2|Q|+1}$ and decide whether its language is empty. $\qquad\square$

4.3   Another Decidable Case in $\Sigma^* \times b^*$

In this part, we complete the proof of Theorem 2.10 by showing the following proposition:

Proposition 4.4 Assume two recognizable languages $\mathbb{R}, \mathbb{P} \subseteq \Sigma^* \times b^*$ such that $\mathbb{P}$ contains some trace of the form $\binom{\lambda}{b^+}$. We can decide whether the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable. $\square$

Proof: We can split $\mathbb{R}$ into $\mathrm{NC}(\mathbb{R})$ and $\mathrm{Cn}(\mathbb{R})$. By Proposition 4.2 and 4.1 (2), it suffices to consider the intersection $\mathrm{NC}(\mathbb{R}) \cap \mathbb{P}^*$. Hence, we assume that $\mathbb{R}$ contains only non-connected traces in the rest of the proof.

By MEZEI's Theorem, we can split $\mathbb{R}$ into finitely many cartesian products and apply Proposition 4.2. Consequently, it suffices to consider the case that $\mathbb{R} = K \times L$ for recognizable languages $K \subseteq \Sigma^*$ and $L \subseteq b^*$. We have $\lambda \notin K \cup L$, because $\mathbb{R}$ contains not any connected trace.

If L is finite, we know by Proposition 4.1 (1) that $\mathbb{R} \cap \mathbb{P}^*$ is recognizable. Hence, it suffices to consider infinite languages L in the rest of the proof.

By Lemma 3.2, we can split L into a finite language and finitely many languages of the form $b^m(b^z)^*$ for some integers $m, z > 0$. By splitting L, we can split $\mathbb{R}$ to use Proposition 4.2, again. Hence, it suffices to consider languages $L = b^m(b^z)^*$ for some integers $m, z > 0$.

We can assume $\binom{\lambda}{b^z} \in \mathbb{P}^*$. If $\binom{\lambda}{b^z} \notin \mathbb{P}^*$, then we proceed as follows: Assume some $n > 0$ such that $\binom{\lambda}{b^n} \in \mathbb{P}$. The language L is the union of the languages $b^{m+jz}(b^{nz})^*$ for $j \in \{0, \ldots, n-1\}$. Then, $\binom{\lambda}{b^{nz}} \in \mathbb{P}^*$. As above, we can split $\mathbb{R}$ by splitting L and use Proposition 4.2.

Now, we transform the language $\mathbb{P}$ into a recognizable language $\mathbb{P}'$ with $\mathbb{R} \cap \mathbb{P}^* = \mathbb{R} \cap \mathbb{P}'^*$ such that $\mathbb{P}'$ satisfies some additional properties.

$$\mathbb{P}' = \mathrm{Cn}(\mathbb{P})^* \mathrm{NC}(\mathbb{P}) \mathrm{Cn}(\mathbb{P})^* \cup \mathrm{NC}(\mathrm{Cn}(\mathbb{P})^*)$$

By Lemma 2.4, both $\mathrm{Cn}(\mathbb{P})$ and $\mathrm{NC}(\mathbb{P})$ are recognizable. Hence, by the closure properties of recognizable trace languages, the language $\mathbb{P}'$ is recognizable.

Of course, we have $\mathbb{P}' \subseteq \mathbb{P}^*$, and thus, $\mathbb{P}'^+ \subseteq \mathbb{P}^*$. Further, every trace in $\mathbb{P}'^+$ is non-connected such that we have $\mathbb{P}'^+ \subseteq \mathrm{NC}(\mathbb{P}^*)$. Assume some trace $t \in \mathrm{NC}(\mathbb{P}^*)$. There is some integer $n > 0$ and traces $t_1, \ldots, t_n \in \mathbb{P}$ with $t_1 \ldots t_n = t$. If $t_1, \ldots, t_n \in \mathrm{Cn}(\mathbb{P})$, then we have $t = t_1 \ldots t_n \in \mathrm{NC}(\mathrm{Cn}(\mathbb{P})^*) \subseteq \mathbb{P}' \subseteq \mathbb{P}'^+$. Otherwise, we have $t \in \mathbb{P}'^k$, where k is the number of non-connected traces among $t_1, \ldots, t_n$. Hence, we have $\mathrm{NC}(\mathbb{P}^*) \subseteq \mathbb{P}'^+$, i.e., we have $\mathrm{NC}(\mathbb{P}^*) = \mathbb{P}'^+$. Because there are only non-connected traces in $\mathbb{R}$ and in particular $\binom{\lambda}{\lambda} \notin \mathbb{R}$, we have $\mathbb{R} \cap \mathbb{P}^* = \mathbb{R} \cap \mathbb{P}'^*$. Consequently, we can decide whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable by deciding whether $\mathbb{R} \cap \mathbb{P}'^*$ is recognizable.

Let $P_0, P_1, \ldots$ be the unique family of languages in Σ^* such that

$$\mathbb{R} \cap \mathbb{P}'^* = \binom{P_0}{b^m} \cup \binom{P_1}{b^{m+z}} \cup \binom{P_2}{b^{m+2z}} \cup \ldots$$

Because every trace in $\mathbb{P}'$ contains the letter b, we have for any integer i

$$\mathbb{R} \cap \mathbb{P}'^{0,\ldots,m+iz} \cap \binom{\Sigma^*}{b^{m+iz}} = \binom{P_i}{b^{m+iz}}$$

Hence, $\binom{P_i}{b^{m+iz}}$ and by MEZEI's Theorem P_i are recognizable for any integer i.

Because $\binom{\lambda}{b^z} \in \mathrm{Cn}(\mathbb{P})^*$, we have $\mathbb{P}'\binom{\lambda}{b^z} \subseteq \mathbb{P}'$. Because $L = b^m(b^z)^*$, we have $\mathbb{R}\binom{\lambda}{b^z} \subseteq \mathbb{R}$. Thus, for every $\binom{u}{v} \in \mathbb{R} \cap \mathbb{P}'^*$, we have $\binom{u}{v}\binom{\lambda}{b^z} \in \mathbb{R} \cap \mathbb{P}'^*$. Hence, we have $P_0 \subseteq P_1 \subseteq P_2 \ldots$ We show the equivalence of four assertions:

1. $\mathbb{R} \cap \mathbb{P}'^*$ is recognizable.

2. There is some integer l such that for $i \geq l$ we have $P_l = P_i$.

Below, we will state assertion (3) and (4). If the integer l in (2) exists, then we have

$$\mathbb{R} \cap \mathbb{P}'^* = \tbinom{P_0}{b^m} \cup \ldots \cup \tbinom{P_{l-1}}{b^{m+(l-1)z}} \cup \tbinom{P_l}{b^{m+lz}(b^z)^*}$$

which is recognizable by MEZEI's Theorem.

Conversely, assume that the integer l in (2) does not exist. Let $i_1, i_2, \ldots$ an infinite sequence of integers such that the languages $P_{i_1}, P_{i_2}, \ldots$ are mutually different. Then, the homomorphism in some automaton for $\mathbb{R} \cap \mathbb{P}'^*$ has to map the traces $\tbinom{\lambda}{b^{m+iz}}$ for $i \in \{i_1, i_2, \ldots\}$ to mutually different states, i.e., any automaton for $\mathbb{R} \cap \mathbb{P}'^*$ has infinitely many states. Hence, such an automaton cannot exist, and thus, $\mathbb{R} \cap \mathbb{P}'^*$ is not recognizable.

In the rest of proof, we consider the decidability of the existence of the integer l in assertion (2). By MEZEI's Theorem, we have $\mathbb{P}' = (K_1 \times L_1) \cup \ldots \cup (K_k \times L_k)$ for some integer k and recognizable languages $K_1, L_1, \ldots, K_k, L_k$. We have $\lambda \notin K_1, L_1, \ldots, K_k, L_k$. By MEZEI's Theorem and Lemma 2.2, we construct automata for $K, K_1, \ldots, K_k$ as follows: We construct a finite monoid P, a surjective homomorphism $g : \Sigma^* \to P$, and subsets $G, G_1, \ldots, G_k \subseteq P$ with $K = g^{-1}(G)$ and $K_i = g^{-1}(G_i)$ for $i \in \{1, \ldots, k\}$. We also construct a finite monoid Q, a surjective homomorphism $h : b^* \to Q$, and sets $F, F_1, \ldots, F_k \subseteq Q$ with $L = h^{-1}(F)$ and $L_i = h^{-1}(F_i)$ for $i \in \{1, \ldots, k\}$.

We construct a distance automaton which is limited in distance iff some integer l in (2) exists. However, at first, we construct some transition automaton $\mathcal{A}$ with (possibly) infinitely many edges as a preliminary tool to explain the idea. Its set of states is $P \times Q$. Its initial state is $\tbinom{\lambda_P}{\lambda_Q}$, where λ_P and λ_Q are the identities in P and Q, respectively. Its set of accepting states are $G \times F$, i.e., the cartesian product of the accepting states of the automata for K and L. For every state $\tbinom{p}{q} \in (P \times Q)$ and every trace $\tbinom{u}{v} \in \mathbb{P}'$, we insert an edge $\left(\tbinom{p}{q}, u, \tbinom{p \cdot g(u)}{q \cdot h(v)} \right)$. Probably, $\mathcal{A}$ has infinitely many edges, i.e., $\mathcal{A}$ is not necessarily a transition automaton. Nevertheless, we can use the terms "path in $\mathcal{A}$", "accepting path in $\mathcal{A}$"... We state assertion (3).

3. There is some integer n such that any word which $\mathcal{A}$ accepts is the label of a path which consists of at most n edges.

Before we show the equivalence (2)$\Leftrightarrow$(3), we show that $\mathcal{A}$ accepts exactly the words which are first components of traces in $\mathbb{R} \cap \mathbb{P}'^*$, i.e., the language of $\mathcal{A}$ is the union $P_0 \cup P_1 \cup \ldots$ Assume some integer i and some $\tbinom{u}{v} \in \mathbb{P}'^i$. Clearly, there is a path in $\mathcal{A}$ from $\tbinom{\lambda_P}{\lambda_Q}$ to $\tbinom{g(u)}{h(v)}$ which consists of i edges and is labeled with u. If additionally $\tbinom{u}{v} \in \mathbb{R}$, then we have $g(u) \in G$ and $h(v) \in F$, and thus, $\mathcal{A}$ accepts u.

Conversely, assume some integer i and some path in $\mathcal{A}$ from $\tbinom{\lambda_P}{\lambda_Q}$ to some state $\tbinom{p}{q}$ which consists of i edges and is labeled with some word u. Then, we have $p = g(u)$ and there is some trace $\tbinom{u}{v} \in \mathbb{P}'^i$ with $h(v) = q$. If additionally $\tbinom{p}{q} \in F \times G$, then $\tbinom{u}{v} \in \mathbb{R}$, and thus, $\tbinom{u}{v} \in \mathbb{R} \cap \mathbb{P}'^i$.

We show (2)$\Rightarrow$(3). Let $n = m + lz$. Assume some word $w \in L(\mathcal{A})$. We have $w \in P_l$, and thus, $\tbinom{w}{b^{m+lz}} \in \mathbb{R} \cap \mathbb{P}'^*$. Because the letter b occurs in every trace in $\mathbb{P}'$, we have $\tbinom{w}{b^{m+lz}} \in \mathbb{P}'^{1, \ldots, m+lz}$. Hence, $\mathcal{A}$ accepts w by a path consisting of at most $m + lz$ edges.

We show (3)$\Rightarrow$(2). Choose some integer l such that $m + lz \geq n|Q|$. Assume some word $w \in P_0 \cup P_1 \ldots$ There is some $n' \leq n$ such that $\mathcal{A}$ accepts w by a path consisting of n' edges. Hence, there are traces $t_1, \ldots, t_{n'} \in \mathbb{P}'$ such that $t_1 \ldots t_{n'} \in \mathbb{R} \cap \mathbb{P}'^*$ and the first component of $t_1 \ldots t_{n'}$ is w. For $i \in \{1, \ldots, n'\}$, we denote $t_i = \tbinom{u_i}{v_i}$. By Lemma 3.1, there is some word $v_i' \in b^*$ such that $|v_i'| < |Q|$ and $h(v_i) = h(v_i')$, for $i \in \{1, \ldots, n'\}$. Let $t_i' = \tbinom{u_i}{v_i'}$. We have $t_1' \ldots t_{n'}' \in \mathbb{R}$, because $h(v_i) = h(v_i')$. The first component of $t_1' \ldots t_{n'}'$ is w. The second component of $t_1' \ldots t_{n'}'$ consists of less than $n'|Q|$ letters, i.e., less than $n|Q|$ letters. Hence, we have $w \in P_l$.

It remains to show the decidability of the existence of the integer n in assertion (3). We construct a distance automaton $\mathcal{A}'$ which is limited in distance iff the integer in assertion (3) exists.

The distance automaton $\mathcal{A}'$ has the same states, initial state, and accepting states as $\mathcal{A}$. It has beside the states of $\mathcal{A}$ some additional states. Assume any two states $\binom{p}{q}, \binom{\hat{p}}{\hat{q}} \in P \times Q$. Above, we inserted probably infinitely many edges between these two states. We examine the set of all edge labels of edges (i.e. paths of length one) between $\binom{p}{q}$ and $\binom{\hat{p}}{\hat{q}}$ in $\mathcal{A}$. We define

$$T = \bigcup_{\substack{\text{for any } p' \in P, \, q' \in Q \text{ with } pp' = \hat{p}, \, qq' = \hat{q}, \\ \text{and } p' \in G_i, q' \in F_i \text{ for some } i \in \{1,...,k\}}} g^{-1}(p')$$

Assume some edge $\left(\binom{p}{q}, u, \binom{\hat{p}}{\hat{q}} \right)$ in $\mathcal{A}$. There is some $v \in b^*$ such that $\binom{u}{v} \in \mathbb{P}'$ and $\binom{\hat{p}}{\hat{q}} = \binom{p \cdot g(u)}{q \cdot h(v)}$. To verify $u \in T$, we set $p' = g(u)$ and $q' = h(v)$. We have $\binom{u}{v} \in \mathbb{P}'$, i.e., there is some $i \in \{1, \ldots, k\}$ with $\binom{u}{v} \in K_i \times L_i$. Then, we have $p' = g(u) \in G_i$ and $q' = h(v) \in F_i$.

Conversely, assume some $u \in T$. Let $p' = g(u)$. Choose some q' which satisfies the properties in the expression for T. Because h is a surjection, there is some $v \in \Sigma^*$ with $h(v) = q'$. There is some $i \in \{1, \ldots, k\}$ with $p' = g(u) \in G_i$ and $q' = h(v) \in F_i$. Hence, $\binom{u}{v} \in (K_i \times L_i) \subseteq \mathbb{P}'$. Thus, there is some edge $\left(\binom{p}{q}, u, \binom{p \cdot g(u)}{q \cdot h(v)} \right)$, i.e., $\left(\binom{p}{q}, u, \binom{\hat{p}}{\hat{q}} \right)$ in $\mathcal{A}$.

Consequently, for any word $u \in \Sigma^*$, there is some edge $\left(\binom{p}{q}, u, \binom{\hat{p}}{\hat{q}} \right)$ in $\mathcal{A}$ iff $u \in T$.

We can construct a transition automaton for T. We construct some transition automaton for T with exactly one accepting state such that the initial state has no incoming edges and the accepting state has no outgoing edges. Further, its edge labels are single letters from Σ. Instead of inserting infinitely many edges between $\binom{p}{q}$ and $\binom{\hat{p}}{\hat{q}}$, we insert the transition automaton for T between these states, i.e., $\binom{p}{q}$ and $\binom{\hat{p}}{\hat{q}}$ are its initial and accepting state. The transition automaton for T simulates the formerly infinitely many edges between $\binom{p}{q}$ and $\binom{\hat{p}}{\hat{q}}$. The edges to $\binom{\hat{p}}{\hat{q}}$ get the distance 1, all other edges get the distance 0.

We proceed this for every pair of states $\binom{p}{q}, \binom{\hat{p}}{\hat{q}} \in P \times Q$. We obtain the distance automaton $\mathcal{A}'$ with the same language as $\mathcal{A}$ but finitely many edges. We state assertion (4).

4. The distance automaton $\mathcal{A}'$ is limited in distance.

We can easily verify (3)$\Leftrightarrow$(4). Moreover, if both (3) and (4) are true, then the least integer n to satisfy (3) is exactly the biggest value $\delta(w)$ for $w \in L(\mathcal{A}') = L(\mathcal{A})$. We can decide by Theorem 3.3 whether assertion (4) is true. Hence, we can decide the recognizability of $\mathbb{R} \cap \mathbb{P}'^*$. $\square$

5 Some Undecidable Cases

5.1 The Problem in $\Sigma^* \times b^*$ in general

Although we worked very hardly in the previous section, there are cases which remained open. Surprisingly, improving Proposition 4.4 by cutting the presumption that some trace of the form $\binom{\lambda}{b^+}$ belongs to $\mathbb{P}$ is not possible, because the problem becomes undecidable.

Within this subsection, we consider an alphabet Γ and picture languages over Γ. We further consider the alphabet $\Sigma = \Gamma \cup \{\#, \&\}$. Assume two integers $n, m \geq 1$ and a picture p over Γ of the size (m, n). A word $w \in \Sigma^*$ represents p iff w consists of the lines of $\hat{p}$ with $\&$ as separators, i.e.,

$$w = \& \#^{n+2} \ \& \# p_{1,1} \ldots p_{1,n} \# \ \& \# p_{2,1} \ldots \ldots p_{m,n} \# \ \& \#^{n+2} \ \&$$

We define some language $\mathbb{K} \subseteq \Sigma^*$ by

$$\mathbb{K} = \& \#^3 \#^* \left(\& \# \Gamma^+ \# \right)^+ \& \#^3 \#^* \&.$$

The language $\mathbb{K}$ is recognizable. The words in $\mathbb{K}$ are not necessarily representations of pictures over Γ, because "the lines can have different lengths".

We call some trace $t \in \Sigma^* \times b^*$ *fair* iff $t = \binom{\& \#^n (\& \# \Gamma^{n-2} \#)^+ \& \#^n \&}{b^n}$ for some $n \geq 3$. The first component of every fair trace represents a picture over Γ. Moreover, for every picture p over Γ there is exactly one fair trace whose first component represents p.

We define a transition automaton $\mathcal{A}$. It has the states start, chk (for check), and acc. The states start and acc are the initial and accepting state, respectively. The instructions (edges) of $\mathcal{A}$ are:

0. $[\text{start}, \binom{\Sigma^* \setminus \mathbb{K}}{b^*}, \text{acc}]$

1. $[\text{start}, \binom{\Sigma^* \&}{\lambda}, \text{chk}]$

2. $[\text{chk}, \binom{\Gamma \cup \#}{b}, \text{chk}]$

3. $[\text{chk}, \binom{\& \Sigma^*}{b^+}, \text{acc}]$

4. $[\text{chk}, \binom{(\Gamma \cup \#) \Sigma^*}{\lambda}, \text{acc}]$

Lemma 5.1 Assume some trace $t \in (\Sigma^* \times b^*)$. The automaton $\mathcal{A}$ accepts t iff t is not fair. $\qquad \square$

Proof: Assume a trace $t \in (\Sigma^* \times b^*)$ which is not fair. If the first component of t does not belong to $\mathbb{K}$, then $\mathcal{A}$ accepts t by instruction (0). Otherwise, there are two words $w_1, w_3 \in \Sigma^*$ and a word $w_2 \in (\Gamma \cup \#)^+$, such that $t = \binom{w_1 \&}{\lambda} \binom{w_2}{v} \binom{\& w_3}{\lambda}$ and $|w_2| \neq |v|$. At first, the automaton parses $\binom{w_1 \&}{\lambda}$ using instruction (1). Then, it uses instruction (2) as many times as possible. Then, depending on whether $|w_2| < |v|$ or $|w_2| > |v|$, it uses instruction (3) or (4), resp., to terminate.

Conversely, assume some trace $t \in (\Sigma^* \times b^*)$ such that $\mathcal{A}$ accepts t. If $\mathcal{A}$ uses instruction (0), then t cannot be fair. Assume $\mathcal{A}$ starts with (1). Then, it uses instruction (2) several times, and it terminates with instruction (3) or (4). After using instruction (1), $\mathcal{A}$ has parsed a trace of the form $\binom{w_1 \&}{\lambda}$ for some $w_1 \in \Sigma^*$. Then, $\mathcal{A}$ uses several times instruction (2). Let $n \in \mathbb{N}$ be the number how often $\mathcal{A}$ uses instruction (2). Thus, it parsed some trace $\binom{w_1 \& w_2}{b^n}$ for some $w_2 \in (\Gamma \cup \#)^n$. After that, the automaton terminates using instruction (3) or (4). If it uses (3), then it parsed a first component with a subword $\& w_2 \&$ with $|w_2| = n$. However, because it used instruction (3) there are more than n letters c in the second component. If it uses instruction (4), then there is some subword $w_2 x \in (\Gamma \cup \{\#\})$ in the first component, but, there are n letters b in the second component. Either way, t is not fair. $\qquad \square$

Now, we extend the automaton $\mathcal{A}$. We assume a local representation Δ over Γ. We extend the automaton $\mathcal{A}$ such that it accepts not only the unfair traces but also the fair traces whose first component encodes a picture which does *not* belong to $L(\Delta)$.

We use one cheap trick. We do not use Δ. Assume a letter $\$ \in \Gamma$ which does not occur in Δ. By Lemma 3.5, we construct a local representation Δ' for the picture language $L(\Delta) \circ \{\$\}^{**}$. The language $L(\Delta')$ is either empty or it contains pictures of arbitrary width. We extend $\mathcal{A}$ such that it accepts the unfair traces and the fair traces whose first component encodes a picture which does *not* belong to $L(\Delta')$.

We define the automaton $\mathcal{A}_{\Delta'}$. It has the same states as $\mathcal{A}$ and additionally, for every two letters $a, c \in \Gamma \cup \#$, the state chk_{ac}. The instructions of $\mathcal{A}_{\Delta'}$ are the instructions of $\mathcal{A}$ and additionally:

5. $[\mathsf{start}, \binom{\Sigma^* ac}{b}, \mathsf{chk}_{ac}]$ for every $a, c \in \Gamma \cup \#$

6. $[\mathsf{chk}_{ac}, \binom{\Sigma}{b}, \mathsf{chk}_{ac}]$ for every $a, c \in \Gamma \cup \#$

7. $[\mathsf{chk}_{ac}, \binom{de\Sigma^*}{\lambda}, \mathsf{acc}]$ for every $a, c, d, e \in \Gamma \cup \#$ with $\left(\begin{smallmatrix} a & c \\ d & e \end{smallmatrix}\right) \notin \Delta'$

Lemma 5.2 Assume some trace $\binom{w}{v} \in (\Sigma^* \times b^*)$. The automaton $\mathcal{A}_{\Delta'}$ accepts $\binom{w}{v}$ iff either

- $\binom{w}{v}$ is not fair, or

- $\binom{w}{v}$ is fair and w encodes some picture $p \in \Gamma^{**}$ with $p \notin L(\Delta')$.   □

Proof: If $\binom{w}{v}$ is not fair, then $\mathcal{A}_{\Delta'}$ accepts $\binom{w}{v}$ as $\mathcal{A}$ accepts this trace (cf. Lemma 5.1).

Let us assume that $\binom{w}{v}$ is fair, and w encodes a picture $p \notin L(\Delta')$. Let (m, n) denote the size of p. Then, we have $v = b^{n+2}$. There are integers i, j with $0 \le i \le m$ and $0 \le j \le n$ such that $\left(\begin{smallmatrix} \hat{p}_{i,j} & \hat{p}_{i,j+1} \\ \hat{p}_{i+1,j} & \hat{p}_{i+1,j+1} \end{smallmatrix}\right) \notin \Delta'$. We denote $\hat{p}_{i,j}$, $\hat{p}_{i,j+1}$, $\hat{p}_{i+1,j}$, and $\hat{p}_{i+1,j+1}$ by a, c, d, and e, respectively.

We factorize w. There are $w_1, w_2, w_3 \in \Sigma^*$ such that $|w_2| = n+1$ and $w = w_1 ac w_2 de w_3$. Hence, $\mathcal{A}_{\Delta'}$ can accept $\binom{w}{v}$ by using instruction (5) $[\mathsf{start}, \binom{w_1 ac}{b}, \mathsf{chk}_{ac}]$, then using instruction (6) $n+1$ times, and finally using (7) $[\mathsf{chk}_{ac}, \binom{de w_3}{\lambda}, \mathsf{acc}]$.

Conversely, assume some trace $\binom{w}{v}$ which $\mathcal{A}_{\Delta'}$ accepts. If $\binom{w}{v}$ is not fair, then we are done. We consider the case that t is fair, i.e., t encodes some picture p. Let (m, n) be the size of p. Then, $v = b^{n+2}$.

The automaton $\mathcal{A}_{\Delta'}$ cannot accept $\binom{w}{v}$ by instruction (0) or by a run starting with (1). Hence, it suffices to consider the case that $\mathcal{A}_{\Delta'}$ accepts $\binom{w}{v}$ by starting with instruction (5). Then, $\mathcal{A}$ accepts $\binom{w}{v}$ by a run using instruction (5) once, several times instruction (6), and once instruction (7). Because $v = b^{n+2}$, it has to use instruction (6) exactly $n+1$ times. Then, there are four letters a, c, d, e from the instructions (5) and (7) in the run of the automaton and there are words $w_1, w_2, w_3 \in \Sigma^*$ such that $w = w_1 ac w_2 de w_3$ and $|w_2| = n + 1$. Hence, the letters a, c, d, e form a subpicture in $\hat{p}$ of size $(2, 2)$ which does not belong to Δ', i.e., $p \notin L(\Delta')$.   □

Now, we can show the following connection:

Proposition 5.3 The language of $\mathcal{A}_{\Delta'}$ is recognizable iff $L(\Delta)$ is empty.   □

Proof: Assume that $L(\Delta)$ is empty. Then, $L(\Delta')$ is also empty. Thus, $\mathcal{A}_{\Delta'}$ accepts the complete monoid $\Sigma^* \times b^*$ which is a recognizable language.

Conversely, assume that $L(\Delta)$ is not empty, but nevertheless, $L(\mathcal{A}_{\Delta'})$ is recognized by the automaton $[Q, h, F]$. Because $L(\Delta') = L(\Delta) \circ \{\$\}^{**}$, the language $L(\Delta')$ contains pictures of arbitrary width. Hence, we can choose $p, s \in L(\Delta')$ with $\mathrm{wdt}(p) \ne \mathrm{wdt}(s)$ such that $h\binom{w_p}{\lambda} = h\binom{w_s}{\lambda}$, where w_p and w_s are the words which encode p and s, respectively. We have $h\binom{w_p}{b^{\mathrm{wdt}(p)+2}} = h\binom{w_s}{b^{\mathrm{wdt}(p)+2}}$. Thus, either both or none of the traces $\binom{w_p}{b^{\mathrm{wdt}(p)+2}}$ and $\binom{w_s}{b^{\mathrm{wdt}(p)+2}}$ belongs to $L(\mathcal{A}_{\Delta'})$. However, $\binom{w_p}{b^{\mathrm{wdt}(p)+2}} \notin L(\mathcal{A}_{\Delta'})$ by Lemma 5.2. On the other hand $\binom{w_s}{b^{\mathrm{wdt}(p)+2}} \in L(\mathcal{A}_{\Delta'})$, because it is not fair.   □

Based on $\mathcal{A}_{\Delta'}$, we define the recognizable language $\mathbb{P}$. We introduce a new letter $k \notin \Sigma$ and consider the monoid $(\Sigma \cup k)^* \times b^*$. We denote by π the homomorphism $\pi \colon \left((\Sigma \cup k)^* \times b^*\right) \to (\Sigma^* \times b^*)$ which erases the letter k. Note that π is a connected homomorphism. Hence, $\pi(T)$ yields a recognizable language for recognizable languages T.

Let $n = |(\Gamma \cup \#)|^2 + 2$. Hence, we can assign for $a, c \in (\Gamma \cup \#)$ the state chk_{ac} and the state chk a number between 1 and $n - 1$.

Now, we define the language $\mathbb{P}$ by defining several languages whose union yields $\mathbb{P}$. We are interested in traces whose first component is of the form $(\Sigma k^*)^* \Sigma$. We distinguish two kinds of these traces: *well-formed* traces which are traces whose first component is $(\Sigma k^n)^* \Sigma$, and *trash*-traces, i.e., traces whose first component belongs to $(\Sigma k^*)^* \Sigma \setminus (\Sigma k^n)^* \Sigma$. We define a so-called trash language $\mathbb{P}_T$.

$$\mathbb{P}_T = \left(\begin{smallmatrix} (\Sigma k^*)^* \Sigma \,\setminus\, (\Sigma k^n)^* \Sigma \\ b^* \end{smallmatrix}\right)$$

The traces in $\mathbb{P}_T$ are not well-formed. Moreover, the concatenation of any trace in $\mathbb{P}_T$ and any other trace yields a trace in $\mathbb{P}_T$, i.e., there are not any well-formed traces in $\left(\begin{smallmatrix} (\Sigma \cup k)^* \\ b^* \end{smallmatrix}\right) \mathbb{P}_T \left(\begin{smallmatrix} (\Sigma \cup k)^* \\ b^* \end{smallmatrix}\right)$.

We define $\mathbb{P}_0$. It consists of well-formed traces.

$$\mathbb{P}_0 = \left\{ \left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) \in \left(\begin{smallmatrix} (\Sigma k^n)^* \Sigma \\ b^* \end{smallmatrix}\right) \;\middle|\; \pi\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) \in \left(\begin{smallmatrix} \Sigma^* \,\setminus\, \mathbb{K} \\ b^* \end{smallmatrix}\right) \right\} = \left(\begin{smallmatrix} (\Sigma k^n)^* \Sigma \\ b^* \end{smallmatrix}\right) \cap \pi^{-1}\left(\begin{smallmatrix} \Sigma^* \,\setminus\, \mathbb{K} \\ b^* \end{smallmatrix}\right)$$

The language $\mathbb{P}_0$ is related to instruction (0) of the automaton. Accordingly, we define $\mathbb{P}_1, \ldots, \mathbb{P}_4$.

$$\mathbb{P}_1 = \left(\begin{smallmatrix} (\Sigma k^n)^* \& k \\ \lambda \end{smallmatrix}\right) \qquad \mathbb{P}_2 = \left(\begin{smallmatrix} k^{n-1}(\Gamma \cup \#)k \\ b \end{smallmatrix}\right) \qquad \mathbb{P}_3 = \left(\begin{smallmatrix} k^{n-1} \& (k^n \Sigma)^* \\ b^+ \end{smallmatrix}\right) \qquad \mathbb{P}_4 = \left(\begin{smallmatrix} k^{n-1}(\Gamma \cup \#)(k^n \Sigma)^* \\ \lambda \end{smallmatrix}\right)$$

Before we continue to define $\mathbb{P}$, we examine the parts of $\mathbb{P}$ which we already defined. Let $\mathbb{P}_{T,\ldots,4}$ denote the union $\mathbb{P}_T \cup \mathbb{P}_0 \cup \ldots \cup \mathbb{P}_4$. We examine the well-formed traces in $\mathbb{P}_{T,\ldots,4}^*$. We cannot obtain a well-formed trace if we concatenate some traces in $\mathbb{P}_{T,\ldots,4}$ and we use a trace in $\mathbb{P}_T$. Moreover, we easily see that the well formed traces in $\mathbb{P}_{T,\ldots,4}^*$ are the traces in $\mathbb{P}_0$ and the traces in $\mathbb{P}_1 \mathbb{P}_2^* (\mathbb{P}_3 \cup \mathbb{P}_4)$. Consequently, we have natural connection between the well-formed traces in $\mathbb{P}_{T,\ldots,4}^*$ and the paths of $\mathcal{A}$. Therefore, if we erase the letter k in some well-formed trace in $\mathbb{P}_{T,\ldots,4}^*$, then we either obtain an unfair trace (cf. Lemma 5.1). Moreover, by applying π on the well-formed traces in $\mathbb{P}_{T,\ldots,4}^*$ we obtain any unfair trace.

Now, we define the remaining parts of $\mathbb{P}$. For every $a, c \in \Gamma \cup \#$, we define three languages $\mathbb{P}_{5,ac}$, $\mathbb{P}_{6,ac}$, and $\mathbb{P}_{7,ac}$. For every $a, c \in \Gamma \cup \#$, we choose some distinct $1 < z < n$.

$$\mathbb{P}_{5,ac} = \left(\begin{smallmatrix} (\Sigma k^n) a k^n c k^z \\ b \end{smallmatrix}\right) \qquad\qquad \mathbb{P}_{6,ac} = \left(\begin{smallmatrix} k^{n-z} \Sigma k^z \\ b \end{smallmatrix}\right)$$

$$\mathbb{P}_{7,ac} = \left\{ \left(\begin{smallmatrix} k^{n-z} d k^z e (k^z \Sigma)^* \\ \lambda \end{smallmatrix}\right) \;\middle|\; d, e \in (\Gamma \cup \#) \text{ with } \left(\begin{smallmatrix} a\ c \\ d\ e \end{smallmatrix}\right) \notin \Delta' \right\}$$

Now, we define $\mathbb{P}$ as the union:

$$\mathbb{P} = \mathbb{P}_T \cup \mathbb{P}_0 \cup \ldots \cup \mathbb{P}_4 \cup \bigcup_{a,c \in (\Gamma \cup \#)} \left(\mathbb{P}_{5,ac} \cup \mathbb{P}_{6,ac} \cup \mathbb{P}_{7,ac}\right)$$

The language $\mathbb{P}$ is recognizable, because it is the union of finitely many recognizable languages. We examine the well-formed traces in $\mathbb{P}^*$. We have

$$\left(\begin{smallmatrix} (\Sigma k^n)^* \Sigma \\ b^* \end{smallmatrix}\right) \cap \mathbb{P}^* \;=\; \mathbb{P}_0 \,\cup\, \mathbb{P}_1 \mathbb{P}_2^* \mathbb{P}_3 \,\cup\, \mathbb{P}_1 \mathbb{P}_2^* \mathbb{P}_4 \,\cup\, \bigcup_{a,c \in (\Gamma \cup \#)} \left(\mathbb{P}_{5,ac} \mathbb{P}_{6,ac}^* \mathbb{P}_{7,ac}\right)$$

Remark 5.4 Consequently, there is a correspondence between the well-formed traces in $\mathbb{P}^*$ and the accepting paths in $\mathcal{A}_{\Delta'}$. For every well-formed trace $\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) \in \mathbb{P}^*$, we have $\pi\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) \in L(\mathcal{A}_{\Delta'})$. Conversely, for every $t \in L(\mathcal{A}_{\Delta'})$, there is some well-formed trace $\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) \in \mathbb{P}^*$ such that $\pi\left(\begin{smallmatrix} u \\ v \end{smallmatrix}\right) = t$.   □

Now, we can show the following connection:

Proposition 5.5 The intersection $\binom{(\Sigma k^*)^* \Sigma}{b+} \cap \mathbb{P}^*$ is recognizable iff $L(\mathcal{A}_{\Delta'})$ is recognizable. □

Proof: We split $\binom{(\Sigma k^*)^* \Sigma}{b+}$ into two recognizable languages by $\binom{(\Sigma k^*)^* \Sigma}{b+} = \mathbb{P}_T \cup \binom{(\Sigma k^n)^* \Sigma}{b+}$ and apply Proposition 4.1. We have $\mathbb{P}_T \cap \mathbb{P}^* = \mathbb{P}_T$, i.e., $\mathbb{P}_T \cap \mathbb{P}^*$ is recognizable. Hence, $\binom{(\Sigma k^*)^* \Sigma}{b+} \cap \mathbb{P}^*$ is recognizable iff the set of the well-formed traces in $\mathbb{P}^*$ is recognizable. Consequently, it suffices to show that the set of the well-formed traces in $\mathbb{P}^*$ is recognizable iff $L(\mathcal{A}_{\Delta'})$ is recognizable.

Assume that $\binom{(\Sigma k^n)^* \Sigma}{b+} \cap \mathbb{P}^*$ is recognizable. By Remark 5.4, we have

$$L(\mathcal{A}_{\Delta'}) = \pi\left(\binom{(\Sigma k^n)^* \Sigma}{b+} \cap \mathbb{P}^*\right)$$

Because π preserves recognizability, $L(\mathcal{A}_{\Delta'})$ is recognizable. Conversely, assume $L(\mathcal{A}_{\Delta'})$ is recognizable. By Remark 5.4, we have

$$\binom{(\Sigma k^n)^* \Sigma}{b+} \cap \mathbb{P}^* = \binom{(\Sigma k^n)^* \Sigma}{b+} \cap \pi^{-1}(L(\mathcal{A}_{\Delta'}))$$

The set $\pi^{-1}(L(\mathcal{A}_{\Delta'}))$ is recognizable because of the closure of recognizable sets under inverse homomorphisms. We immediately see that $\binom{(\Sigma k^n)^* \Sigma}{b+} \cap \mathbb{P}^*$ is recognizable. □

From Theorem 3.4, Lemma 5.3, Proposition 5.4, we obtain the following corollary:

Corollary 5.6 Assume some alphabet Σ and two letters $b, k \notin \Sigma$. It is not decidable whether for a recognizable language $\mathbb{P} \subseteq (\Sigma \cup k)^* \times b^*$, the intersection $\binom{(\Sigma k^*)^* \Sigma}{b^*} \cap \mathbb{P}^*$ is recognizable. □

Finally, we boil down this result to P3.

Theorem 5.7 It is not decidable whether for some recognizable language $\mathbb{P} \subseteq \{a, c\}^* \times b^*$, the intersection $\binom{(ac^*)^* a}{b^*} \cap \mathbb{P}^*$ is recognizable. □

Proof: Assume such an algorithm. Then, we can contradict Corollary 5.6. Assume Σ, b, k, and $\mathbb{P}$ as in Corollary 5.6. We show how to decide whether $\binom{(\Sigma k^*)^* \Sigma}{c^*} \cap \mathbb{P}^*$ is recognizable. Let $h : (\Sigma \cup k)^* \to \{a, c\}^*$ be an injective homomorphism with $h(k) \in (ca^*)^* c$ and $h(\Sigma) \subseteq (ac^*)^* a$. We extend h to an injective and connected homomorphism $h : (\Sigma \cup k)^* \times b^* \to \{a, c\}^* \times b^*$ by setting $h\binom{\lambda}{b} = \binom{\lambda}{b}$. Then, $\binom{(\Sigma k^*)^* \Sigma}{b^*} \cap \mathbb{P}^*$ is recognizable iff $h\left(\binom{(\Sigma k^*)^* \Sigma}{b^*} \cap \mathbb{P}^*\right)$ is recognizable. We have

$$h\left(\binom{(\Sigma k^*)^* \Sigma}{b^*} \cap \mathbb{P}^*\right) = h\binom{(\Sigma k^*)^* \Sigma}{b^*} \cap h(\mathbb{P})^* = \binom{(ac^*)^* a}{b^*} \cap h(\mathbb{P})^*$$

We can decide recognizability of the last set by the assumed algorithm. □

5.2 The Problem in $\Sigma^* \times \Gamma^*$

In this part, we consider the problem whether $\mathbb{R} \cap \mathbb{P}^*$ is recognizable for recognizable languages in trace monoids of the form $\Sigma^* \times \Gamma^*$. We assume a PCP-instance consisting of alphabets Δ and Σ and biprefix homomorphisms $\alpha, \beta : \Delta^* \to \Sigma^*$. We assume $|\Sigma| = 2$. Let l be an integer such that for every $i \in \Delta$, we have $|\alpha(i)| < l$ and $|\beta(i)| < l$.

In our constructions, below, it will be more convenient to consider monoids $\Sigma^* \times \Sigma^*$. Note that the monoid $\Sigma^* \times \Sigma^*$ is isomorphic to a trace monoid $\Sigma^* \times \Gamma^*$ where Γ is any disjoint copy of Σ.

Similar to the previous part, we construct some automaton step by step. At first, we consider an automaton $\mathcal{A}$ with the states start, loop_1, and err. The states start and err are the initial and accepting state, respectively. The instructions (edges) of $\mathcal{A}$ are:

    0. $[\mathsf{start}, \binom{\lambda}{\lambda}, \mathsf{loop}_1]$

    1. $[\mathsf{loop}_1, \binom{a}{a}, \mathsf{loop}_1]$ for every $a \in \Sigma$

    2. $[\mathsf{loop}_1, \binom{a}{b}, \mathsf{err}]$ for every $a, b \in \Sigma$ with $a \neq b$

    3. $[\mathsf{err}, \binom{a}{\lambda}, \mathsf{err}]$ and $[\mathsf{err}, \binom{\lambda}{a}, \mathsf{err}]$ for every $a \in \Sigma$

Lemma 5.8 Assume some trace $\binom{u}{v} \in (\Sigma^* \times \Sigma^*)$. The automaton $\mathcal{A}$ accepts $\binom{u}{v}$ iff $u \not\sqcap v$. $\square$

Proof: Assume that u and v are not prefix consistent, i.e., there are $x, u', v' \in \Sigma^*$ and $a \neq b \in \Sigma$ such that $\binom{u}{v} = \binom{xau'}{xbv'}$. Then, $\mathcal{A}$ accepts $\binom{u}{v}$ by instruction (0), several times instruction (1) to parse $\binom{x}{x}$, instruction (2) to parse $\binom{a}{b}$, and several times instruction (3) to parse $\binom{u'}{v'}$.

    Conversely, if $u \sqcap v$, then $\mathcal{A}$ is forced to use instruction (0) and several times instruction (1), i.e., it remains in the state loop_1. $\square$

We define the automaton $\mathcal{A}'$ by adding a state loop_2 and some edges to $\mathcal{A}$:

    4. $[\mathsf{start}, \binom{\lambda}{\lambda}, \mathsf{loop}_2]$

    5. $[\mathsf{loop}_2, \binom{\alpha(i)}{\beta(i)}, \mathsf{loop}_2]$ for every $i \in \Delta$

    6. $[\mathsf{loop}_2, \binom{u}{v}, \mathsf{err}]$ for $u, v \in \Sigma^{1,\dots,l}$ with $\binom{u}{v} \not\sqcap \binom{\alpha(i)}{\beta(i)}$ for every $i \in \Delta$

Lemma 5.9 The automaton $\mathcal{A}'$ accepts some trace $\binom{u}{v} \in (\Sigma^* \times \Sigma^*)$ iff one of the following conditions is true:

(A) $u \not\sqcap v$,

(B) for every $w \in \Delta^*$ with $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{u}{v}$, we have $\alpha(w) \sqsubset u$ and $\beta(w) \sqsubset v$. $\square$

Proof: At first, we show that if u and v satisfy (A) or (B), then $\mathcal{A}$ accepts $\binom{u}{v}$. If u and v satisfy condition (A), then $\mathcal{A}'$ accepts $\binom{u}{v}$ as $\mathcal{A}$ does (cf. Lemma 5.8).

    Assume that u and v satisfy (B). Let $w \in \Delta^*$ be the longest word with $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{u}{v}$. The word w is unique, because α and β are biprefix homomorphisms.

    There are non-empty words $u', v' \in \Sigma^+$ such that $\binom{u}{v} = \binom{\alpha(w)u'}{\beta(w)v'}$. Assume there is an $i \in \Delta$ such that $\binom{\alpha(i)}{\beta(i)} \sqcap \binom{u'}{v'}$. Then, we also have $\binom{\alpha(wi)}{\beta(wi)} \sqcap \binom{u}{v}$. If $u' \sqsubset \alpha(i)$ or $v' \sqsubset \beta(i)$, then we have $u \sqsubset \alpha(wi)$ or $v \sqsubset \beta(wi)$ which contradicts condition (B). Hence, we have $\alpha(i) \sqsubseteq u'$ and $\beta(i) \sqsubseteq v'$. Then, we have $\binom{\alpha(wi)}{\beta(wi)} \sqsubseteq \binom{u}{v}$ which contradicts the choice of the longest word w. Thus, for every $i \in \Delta$, we have $\binom{\alpha(i)}{\beta(i)} \not\sqcap \binom{u'}{v'}$.

    Above, we defined an integer l such that $|\alpha(i)| < l$ and $|\beta(i)| < l$ for $i \in \Delta$. We factorize u' into $u_1, u_2 \in \Sigma^*$ such that $|u_1| = \min\{|u'|, l\}$. Accordingly, we factorize v' into $v_1, v_2 \in \Sigma^*$. We have $\binom{u}{v} = \binom{\alpha(w)u_1u_2}{\beta(w)v_1v_2}$. Because for every $i \in \Delta$, we have $\binom{\alpha(i)}{\beta(i)} \not\sqcap \binom{u'}{v'}$, we also have $\binom{\alpha(i)}{\beta(i)} \not\sqcap \binom{u_1}{v_1}$. Consequently, there is some instruction (6) $[\mathsf{loop}_2, \binom{u_1}{v_1}, \mathsf{err}]$ in $\mathcal{A}'$.

    Now, it is immediate that $\mathcal{A}'$ accepts $\binom{u}{v} = \binom{\alpha(w)u_1u_2}{\beta(w)v_1v_2}$. It uses instruction (4) and $|w|$ times instruction (5) to parse $\binom{\alpha(w)}{\beta(w)}$. Then, it uses instruction (6) to parse $\binom{u_1}{v_1}$ and several times instruction (3) parse $\binom{u_2}{v_2}$ and to terminate.

    Conversely, assume words $u, v \in \Sigma^*$ such that $\mathcal{A}'$ accepts $\binom{u}{v}$. If $\mathcal{A}'$ accepts $\binom{u}{v}$ by run which uses just the instructions (0) to (3), then u and v are not prefix consistent. Hence, we just have to consider the case that $\mathcal{A}'$ accepts $\binom{u}{v}$ by a run $(4)(5)^*(6)(3)^*$. Therefore, we have $\binom{u}{v} = \binom{\alpha(z)u_1u_2}{\beta(z)v_1v_2}$

for some $z \in \Delta^*$, $u_1, v_1 \in \Sigma^+$, and $u_2, v_2 \in \Sigma^*$, and $\mathcal{A}'$ parsed $\binom{\alpha(z)}{\beta(z)}$, $\binom{u_1}{v_1}$, and $\binom{u_2}{v_2}$ by instructions (5) ($|z|$ times), instruction (6), and (3) respectively.

We show that u and v satisfy condition (B). Assume some word $w \in \Delta^*$ with $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{\alpha(z)u_1u_2}{\beta(z)v_1v_2}$. If $|w| \leq |z|$, then we have $w \sqsubseteq z$, because α and β are biprefix homomorphisms. Then, we have $\alpha(w) \sqsubseteq \alpha(z) \sqsubset \alpha(z)u_1u_2 = u$ and $\beta(w) \sqsubseteq \beta(z) \sqsubset \beta(z)v_1v_2 = v$, i.e., (B) is verified. So assume that $|z| < |w|$. We have $z \sqsubset w$. Choose the $i \in \Delta$ such that $zi \sqsubseteq w$. From $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{\alpha(z)u_1u_2}{\beta(z)v_1v_2}$, we have $\binom{\alpha(zi)}{\beta(zi)} \sqcap \binom{\alpha(z)u_1u_2}{\beta(z)v_1v_2}$. Then, we also have $\binom{\alpha(i)}{\beta(i)} \sqcap \binom{u_1u_2}{v_1v_2}$ and $\binom{\alpha(i)}{\beta(i)} \sqcap \binom{u_1}{v_1}$. Thus, $\mathcal{A}'$ cannot parse $\binom{u_1}{v_1}$ by instruction (6) as we assumed, above. Consequently, there is not any word $w \in \Sigma^*$ with $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{u}{v}$ and $|z| < |w|$. $\qquad\square$

Lemma 5.10 If the PCP instance has no infinite solution, then there is some integer n such that $\mathcal{A}'$ accepts every trace $\binom{u}{v} \in \Sigma^* \times \Sigma^*$ with $|u| \geq n$ and $|v| \geq n$. $\qquad\square$

Proof: If the PCP instance has no infinite solution, then there are only finitely many words $w \in \Delta^*$ such that $\alpha(w) \sqcap \beta(w)$ (cf. Lemma 3.7). Let n' be an integer such that for every $w \in \Delta^*$ with $|w| \geq n'$, we have $\alpha(w) \not\sqcap \beta(w)$. Let $n = n'l$.

Assume words $u, v \in \Sigma^*$ with $|u| \geq n$ and $|v| \geq n$. If $u \not\sqcap v$, then $\mathcal{A}'$ accepts $\binom{u}{v}$ (cf. condition (A) of Lemma 5.9). Assume $u \sqcap v$. We show that u and v satisfy condition (B) in Lemma 5.9. Assume some $w \in \Delta^*$ such that $\binom{\alpha(w)}{\beta(w)} \sqcap \binom{u}{v}$. If $|w| < n'$, then $|\alpha(w)| < n$. Because $|u| \geq n$, we have $\alpha(w) \sqsubset u$. Accordingly, we have $\beta(w) \sqsubset v$. Assume $|w| \geq n'$. Let w' be the prefix of w with $|w'| = n'$. Then, we have $\alpha(w') \not\sqcap \beta(w')$. We have $\binom{\alpha(w')}{\beta(w')} \sqcap \binom{u}{v}$. We have $|\alpha(w')| \leq n$ and $n \leq |u|$, i.e., we have $\alpha(w') \sqsubseteq u$. Accordingly, we obtain $\beta(w') \sqsubseteq v$. This contradicts that u and v are prefix consistent. Consequently, u and v satisfy condition (B) in Lemma 5.9. $\qquad\square$

Lemma 5.11 If the PCP instance has a an infinite solution, then there is an infinite sequence of words $u_1 \sqsubset u_2 \sqsubset \ldots \in \Sigma^*$ and for every integers $0 < i < j$ there is some word v such that $\mathcal{A}'$ does not accept $\binom{u_i}{v}$, but $\mathcal{A}'$ accepts $\binom{u_j}{v}$. $\qquad\square$

Proof: Let $i_1i_2\ldots$ be an infinite solution. We choose a sequence $w_1 \sqsubset w_2 \ldots \in \Delta^*$ of prefixes of $i_1i_2\ldots$ such that for every $i > 0$, we have $|\beta(w_i)| < |\alpha(w_{i+1})|$, i.e., we have $\beta(w_i) \sqsubset \alpha(w_{i+1})$. We set for $i > 0$, $u_i = \alpha(w_i)$. Then, we have $u_1 \sqsubset u_2 \sqsubset \ldots$

Assume some $0 < i < j$. We show the existence of the desired word $v \in \Sigma^*$. Let z be the longer word of $\alpha(w_i)$ and $\beta(w_i)$. We have $\alpha(w_i) \sqsubseteq z$, $\beta(w_i) \sqsubseteq z$, and $z \sqsubset \alpha(w_j)$. Let $a \in \Sigma$ be a letter such that $\alpha(w_j) \not\sqcap za$. Let $v = za$.

The trace $\binom{u_i}{v} = \binom{\alpha(w_i)}{za}$ does not satisfy condition (A) in Lemma 5.9, because $\alpha(w_i) \sqsubseteq z \sqsubset za$. It does not satisfy (B), because we have $\binom{\alpha(w_i)}{\beta(w_i)} \sqcap \binom{\alpha(w_i)}{za}$ but we have not $\alpha(w_i) \sqsubset \alpha(w_i)$. Hence, $\mathcal{A}'$ does not accept $\binom{u_i}{v_i} = \binom{\alpha(w_i)}{za}$. However, it accepts $\binom{u_j}{v_i} = \binom{\alpha(w_j)}{za}$, because $\alpha(w_j) \not\sqcap za$. $\qquad\square$

Now, we can define suitable languages... We enrich Σ by new letters s, l_1, l_2, and e which stand for start, loop_1, loop_2, and err, respectively. We set $\Gamma = \Sigma \cup \{\mathsf{s}, \mathsf{l}_1, \mathsf{l}_2, \mathsf{e}\}$, and examine recognizable languages in $\Gamma^* \times \Gamma^*$. We define

$$\mathbb{R} = \binom{(\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\Sigma)^+\mathsf{e}}{(\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\Sigma)^+\mathsf{e}} \subseteq \Gamma^* \times \Gamma^*$$

We further define finite languages $\mathbb{P}_0, \ldots, \mathbb{P}_6 \subseteq \Gamma^* \times \Gamma^*$ which correspond to the instructions (0) to (6) of $\mathcal{A}'$. Let $\mathbb{P} = \mathbb{P}_0 \cup \ldots \cup \mathbb{P}_6$.

$$\mathbb{P}_0 = \left\{ \binom{\mathsf{e}\mathsf{l}_1}{\mathsf{e}\mathsf{l}_1} \right\}$$

$$\mathbb{P}_1 = \left\{ \binom{\mathsf{l}_2\mathsf{s}a\mathsf{e}\mathsf{l}_1}{\mathsf{l}_2\mathsf{s}a\mathsf{e}\mathsf{l}_1} \;\middle|\; a \in \Sigma \right\}$$

$$\mathbb{P}_2 = \left\{ \left(\begin{smallmatrix}\mathsf{l}_2\mathsf{s}a\mathsf{e}\\ \mathsf{l}_2\mathsf{s}b\mathsf{e}\end{smallmatrix}\right) \;\middle|\; a,b \in \Sigma,\ a \neq b \right\}$$

$$\mathbb{P}_3 = \left\{ \left(\begin{smallmatrix}\mathsf{l}_1\mathsf{l}_2\mathsf{s}a\mathsf{e}\\ \lambda\end{smallmatrix}\right) \;\middle|\; a \in \Sigma \right\} \cup \left\{ \left(\begin{smallmatrix}\lambda\\ \mathsf{l}_1\mathsf{l}_2\mathsf{s}a\mathsf{e}\end{smallmatrix}\right) \;\middle|\; a \in \Sigma \right\}$$

$$\mathbb{P}_4 = \left\{ \left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\\ \mathsf{e}\mathsf{l}_1\mathsf{l}_2\end{smallmatrix}\right) \right\}$$

To define $\mathbb{P}_5$ and $\mathbb{P}_6$, we introduce a mapping $\gamma : \Sigma^+ \to \Gamma^+$. For any $a \in \Sigma$ and any $w \in \Sigma^+$, we define $\gamma(a) = a$ and $\gamma(wa) = \gamma(w)\,\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\,a$. For instance, we have $\gamma(abc) = a\,\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\,b\,\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\,c$.

$$\mathbb{P}_5 = \left\{ \left(\begin{smallmatrix}\mathsf{s}\gamma(\alpha(i))\mathsf{e}\mathsf{l}_1\mathsf{l}_2\\ \mathsf{s}\gamma(\beta(i))\mathsf{e}\mathsf{l}_1\mathsf{l}_2\end{smallmatrix}\right) \;\middle|\; i \in \Delta \right\}$$

$$\mathbb{P}_6 = \left\{ \left(\begin{smallmatrix}\mathsf{s}\gamma(u)\mathsf{e}\\ \mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right) \;\middle|\; u,v \in \Sigma^{1,\dots,l} \text{ with } \left(\begin{smallmatrix}u\\v\end{smallmatrix}\right) \not\sqsubseteq \left(\begin{smallmatrix}\alpha(i)\\\beta(i)\end{smallmatrix}\right) \text{ for every } i \in \Delta \right\}$$

There is an obvious correspondence between the traces in $\mathbb{R} \cap \mathbb{P}^*$ and the runs of $\mathcal{A}'$. It accepts some trace $\left(\begin{smallmatrix}u\\v\end{smallmatrix}\right) \in (\Sigma^* \times \Sigma^*)$ iff $\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u)\mathsf{e}\\\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right) \in \mathbb{R} \cap \mathbb{P}^*$. Hence, we obtain the following proposition:

Proposition 5.12 The set $\mathbb{R} \cap \mathbb{P}^*$ is recognizable iff the PCP instance has no infinite solution. $\square$

Proof: Assume that the PCP instance has no infinite solution. By Lemma 5.10, there is some n such that $\mathcal{A}'$ accepts any trace $\left(\begin{smallmatrix}u\\v\end{smallmatrix}\right) \in (\Sigma^* \times \Sigma^*)$ if both $|u| \geq n$ and $|v| \geq n$. To show recognizability of $\mathbb{R} \cap \mathbb{P}^*$, we split $\mathbb{R}$ into three recognizable languages

$$\mathbb{R}_1 = \mathbb{R} \cap \left(\begin{smallmatrix}\Gamma^{0,\dots,5n}\\\Gamma^*\end{smallmatrix}\right) \qquad \mathbb{R}_2 = \mathbb{R} \cap \left(\begin{smallmatrix}\Gamma^*\\\Gamma^{0,\dots,5n}\end{smallmatrix}\right) \qquad \mathbb{R}_3 = \mathbb{R} \cap \left(\begin{smallmatrix}\Gamma^{5n+1}\Gamma^*\\\Gamma^{5n+1}\Gamma^*\end{smallmatrix}\right)$$

By Proposition 4.2 and 4.1, it suffices to show that $\mathbb{R}_3 \cap \mathbb{P}^*$ is recognizable. We show recognizability of $\mathbb{R}_3 \cap \mathbb{P}^*$ by showing $\mathbb{R}_3 \cap \mathbb{P}^* = \mathbb{R}_3$, i.e., we show $\mathbb{R}_3 \subseteq \mathbb{P}^*$. Indeed, every trace in $\mathbb{R}_3$ is of the form $\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u)\mathsf{e}\\\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right)$ for some $u,v \in \Sigma^*$ with $|u| \geq n$ and $|v| \geq n$. By Lemma 5.10, $\mathcal{A}'$ accepts $\left(\begin{smallmatrix}u\\v\end{smallmatrix}\right)$, and thus, we have $\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u)\mathsf{e}\\\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right) \in \mathbb{P}^*$. Consequently, $\mathbb{R}_3 \subseteq \mathbb{P}^*$.

Conversely, assume that the PCP instance has an infinite solution, but nevertheless, $\mathbb{R} \cap \mathbb{P}^*$ is recognized by some automaton $[Q, h, F]$. Let $u_1 \sqsubset u_2 \dots$ be an infinite sequence as in Lemma 5.11. We choose two integers $0 < i < j$ such that $h\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}\\\lambda\end{smallmatrix}\right) = h\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_j)\mathsf{e}\\\lambda\end{smallmatrix}\right)$. Then, for every word $v \in \Sigma^+$, the automaton $[Q, h, F]$ accepts either none or both of the traces $\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}\\\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right)$ and $\left(\begin{smallmatrix}\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_j)\mathsf{e}\\\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}\end{smallmatrix}\right)$. Hence, for any $v \in \Sigma^+$, $\mathcal{A}'$ accepts either both or none of the traces $\left(\begin{smallmatrix}u_i\\v\end{smallmatrix}\right)$ and $\left(\begin{smallmatrix}u_j\\v\end{smallmatrix}\right)$. This contradicts Lemma 5.11. $\square$

Proof of Theorem 2.8(2): By Proposition 5.12 and Theorem 3.6, it is undecidable whether for some finite language $\mathbb{P} \subseteq (\Gamma^* \times \Gamma^*)$ the intersection $\mathbb{R} \cap \mathbb{P}^*$ is recognizable. By choosing some connected and injective homomorphism from $\Gamma^* \times \Gamma^*$ to C4, we obtain Theorem 2.8(2). $\square$

Remark 5.13 For instance, we can use the homomorphism $h : (\Gamma^* \times \Gamma^*) \to (\{a,c\}^* \times \{b,d\}^*)$ which maps $\left(\begin{smallmatrix}\mathsf{e}\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}\mathsf{l}_1\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}\mathsf{l}_2\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}\mathsf{s}\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}a\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}b\\\lambda\end{smallmatrix}\right)$ to $\left(\begin{smallmatrix}aa\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}aba\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}abb\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}ba\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}bba\\\lambda\end{smallmatrix}\right)$, $\left(\begin{smallmatrix}bbb\\\lambda\end{smallmatrix}\right)$, and the traces in $\lambda \times \Gamma$ similarly to traces in $\left(\begin{smallmatrix}\lambda\\\{b,d\}^*\end{smallmatrix}\right)$. $\square$

We make some slight modifications to prove Theorem 2.8(3). We define so-called trash languages

$$P_T = \bigcup_{x,y \in \Gamma,\ xy \notin \{\mathsf{e}\mathsf{l}_1,\,\mathsf{l}_1\mathsf{l}_2,\,\mathsf{l}_2\mathsf{s},\,\mathsf{s}\Sigma,\,\Sigma\mathsf{e}\}} \Gamma^* xy \Gamma^* \qquad\qquad \mathbb{P}_T = \left(\begin{smallmatrix}\Gamma^*\\P_T\end{smallmatrix}\right) \cup \left(\begin{smallmatrix}P_T\\\Gamma^*\end{smallmatrix}\right)$$

Clearly, P_T and $\mathbb{P}_T$ are infinite recognizable languages in Γ^* and $\Gamma^* \times \Gamma^*$, respectively. We define another language

$$\mathbb{P}' = \mathbb{P} \cup \mathbb{P}_T \cup \left\{ \binom{\mathsf{l}_1}{\lambda}, \binom{\lambda}{\mathsf{l}_1}, \binom{\mathsf{l}_1\mathsf{l}_2}{\lambda}, \binom{\lambda}{\mathsf{l}_1\mathsf{l}_2} \right\}$$

Now, we can show the following Proposition:

Proposition 5.14 The intersection $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$ is recognizable iff the PCP instance has no infinite solution. $\square$

Proof: At first, we assume that the PCP instance has an infinite solution, but $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$ is recognized by some automaton $[Q, h, F]$. There is some infinite sequence $u_1 \sqsubset u_2 \sqsubset \ldots$ by Lemma 5.11. We choose $0 < i < j$ such that $h\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\lambda} = h\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_j)\mathsf{e}}{\lambda}$. By Lemma 5.11, there is some word $v \in \Sigma^*$ such that $\mathcal{A}'$ accepts $\binom{u_j}{v}$, but it does not accept $\binom{u_i}{v}$. Because $\mathcal{A}'$ accepts $\binom{u_j}{v}$, we have $\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_j)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}} \in (\mathbb{R} \cap \mathbb{P}^*) \subseteq (\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*)$. We have chosen i and j such that $h\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\lambda} = h\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_j)\mathsf{e}}{\lambda}$. Consequently, we have $\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}} \in (\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*)$.

We examine some factorization of $\binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}}$ into traces from $\mathbb{P}'$. There are some integer k and traces $t_1, \ldots, t_k \in \mathbb{P}'$ such that $t_1 \ldots t_k = \binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}}$. Clearly, $t_1, \ldots, t_k \notin \mathbb{P}_T$. We choose the biggest integer $k' \leq k$ such that $t_1, \ldots, t_{k'} \in \mathbb{P}$. We have $t_1, t_2 \in \mathbb{P}$, i.e., $k' \geq 2$.

Now, we show that we have $t_1 \ldots t_{k'} \in (\mathbb{R} \cap \mathbb{P}^*)$. It suffices to show $t_1 \ldots t_{k'} \in \mathbb{R}$. If $k' = k$, then we have $t_1 \ldots t_{k'} = t_1 \ldots t_k = \binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}} \in \mathbb{R}$. So assume $k' < k$. We examine $t_1 \ldots t_{k'}$. We have $t_{k'+1} \in \left\{ \binom{\mathsf{l}_1}{\lambda}, \binom{\lambda}{\mathsf{l}_1}, \binom{\mathsf{l}_1\mathsf{l}_2}{\lambda}, \binom{\lambda}{\mathsf{l}_1\mathsf{l}_2} \right\}$. Consequently, one of the two last letters of $t_1 \ldots t_{k'}$ is the letter e, i.e., the first or the second component of $t_1 \ldots t_{k'}$ ends with the letter e. By an induction on t_1, $t_1 t_2$, $\ldots$, $t_1 t_2 \ldots t_{k'}$, we can show that the first and the second component of $t_1 \ldots t_{k'}$ end with the same letter. Consequently, both components of $t_1 \ldots t_{k'}$ end with the letter e. Further, $t_1 \ldots t_{k'} \sqsubseteq t_1 \ldots t_k = \binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u_i)\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v)\mathsf{e}}$. Thus, $t_1 \ldots t_{k'} \in \mathbb{R}$.

Consequently, there are $u', v' \in \Sigma^*$ with $\binom{u'}{v'} \sqsubseteq \binom{u_i}{v}$ such that we have $t_1 \ldots t_{k'} = \binom{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(u')\mathsf{e}}{\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\gamma(v')\mathsf{e}}$. Because $t_1 \ldots t_{k'} \in (\mathbb{R} \cap \mathbb{P}^*)$, the automaton $\mathcal{A}'$ accepts the trace $\binom{u'}{v'}$, i.e., it can reach the state err by reading $\binom{u'}{v'}$. Because $\binom{u'}{v'} \sqsubseteq \binom{u_i}{v}$, the automaton $\mathcal{A}'$ also accepts $\binom{u_i}{v}$. This is a contradiction.

Conversely, assume that the PCP instance has no infinite solution. By Lemma 5.10, there is some n such that $\mathcal{A}'$ accepts any trace $\binom{u}{v} \in (\Sigma^* \times \Sigma^*)$ if both $|u| \geq n$ and $|v| \geq n$. To show recognizability of $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$, we split $\binom{\mathsf{e}\Gamma^*}{\Gamma^*}$ into recognizable languages

$$\mathbb{R}_0 = \binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}_T$$

$$\mathbb{R}_1 = \binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \binom{\Gamma^{0,\ldots,5n}}{\Gamma^*}$$

$$\mathbb{R}_2 = \binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \binom{\Gamma^*}{\Gamma^{0,\ldots,5n}}$$

$$\mathbb{R}_3 = \left(\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \setminus \mathbb{P}_T \right) \cap \binom{\mathsf{e}\Gamma^*\{\mathsf{e}, \mathsf{l}_1, \mathsf{l}_2\}}{\mathsf{e}\Gamma^*\{\mathsf{e}, \mathsf{l}_1, \mathsf{l}_2\}}$$

$$\mathbb{R}_4 = \binom{\mathsf{e}\Gamma^*}{\Gamma^*} \setminus \mathbb{P}_T \setminus \mathbb{R}_1 \setminus \mathbb{R}_2 \setminus \mathbb{R}_3$$

By Proposition 4.2, we can show recognizability of $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$ by showing recognizability of $\mathbb{R}_i \cap \mathbb{P}'^*$ for $i \in \{0, \ldots, 4\}$. We have $\mathbb{R}_0 \subseteq \mathbb{P}_T \subseteq \mathbb{P}' \subseteq \mathbb{P}'^*$, and thus, $\mathbb{R}_0 \cap \mathbb{P}'^*$ yields $\mathbb{R}_0$ which is recognizable. Recognizability of $\mathbb{R}_1 \cap \mathbb{P}'^*$ and $\mathbb{R}_2 \cap \mathbb{P}'^*$ follows from Proposition 4.1.

We show that $\mathbb{R}_4 \cap \mathbb{P}'^* = \emptyset$. Assume some $t \in (\mathbb{R}_4 \cap \mathbb{P}'^*)$. Traces from $\mathbb{P}_T$ cannot occur in factorizations of t, otherwise we have $t \in \mathbb{P}_T$, i.e., $t \notin \mathbb{R}_4$. The second component of t is non-empty, otherwise $t \in \mathbb{R}_2$ and $t \notin \mathbb{R}_4$.

Assume that the last letter of first or second component of t is some letter from $\Sigma \cup \mathsf{s}$. Then, there is some trace from $\mathbb{P}_T$ in every factorization of t into traces of $\mathbb{P}'^*$. This implies $t \in \mathbb{P}_T$ and $t \notin \mathbb{R}_4$. Thus, the last letters of both components of t are e, l_1, or l_2.

Assume that the first letter of the second component of t is the letter e. Then, we have $t \in \mathbb{P}_T$ or $t \in \mathbb{R}_3$. Either way, $t \notin \mathbb{R}_4$. Hence, the first letter of the second component of t is different from e. Hence, any factorization of t into traces from $\mathbb{P}'$ has to start with several times $\binom{\lambda}{\mathsf{l}_1}$ or $\binom{\lambda}{\mathsf{l}_1 \mathsf{l}_2}$, and to continue with some trace from $\mathbb{P}_0$ or $\mathbb{P}_4$. However, two or more traces of $\binom{\lambda}{\mathsf{l}_1}$ or $\binom{\lambda}{\mathsf{l}_1 \mathsf{l}_2}$ at the beginning would imply $t \in \mathbb{P}_T$ which is a contradiction. Hence, any factorization of t starts with exactly one of the traces $\binom{\lambda}{\mathsf{l}_1}$ and $\binom{\lambda}{\mathsf{l}_1 \mathsf{l}_2}$ followed by some trace from $\mathbb{P}_0$ or $\mathbb{P}_4$. However, this yields a subword $\mathsf{l}_1 \mathsf{e}$ or $\mathsf{l}_2 \mathsf{e}$ in the second component of t and implies $t \in \mathbb{P}_T$. Consequently, it is not possible to factorize t into traces from $\mathbb{P}'$. Thus, t does not exist, i.e., $\mathbb{R}_4 \cap \mathbb{P}'^* = \emptyset$.

We show $\mathbb{R}_3 \cap \mathbb{P}'^* = \mathbb{R}_3$. Assume some $t \in \mathbb{R}_3$. Both components of t start with the letter e and end with e, l_1, or l_2. Further, we have $t \notin \mathbb{P}_T$ and both components of t are longer than $5n + 3$ letters. Consequently, we can factorize t as $t = t_1 t_2 t_3$, where $t_1 \in \mathbb{R} = \binom{(\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\Sigma)^+ \mathsf{e}}{(\mathsf{e}\mathsf{l}_1\mathsf{l}_2\mathsf{s}\Sigma)^+ \mathsf{e}}$ (as above), $t_1 \in \{\binom{\lambda}{\lambda}, \binom{\mathsf{l}_1}{\lambda}, \binom{\mathsf{l}_1 \mathsf{l}_2}{\lambda}\}$, and $t_2 \in \{\binom{\lambda}{\lambda}, \binom{\lambda}{\mathsf{l}_1}, \binom{\lambda}{\mathsf{l}_1 \mathsf{l}_2}\}$. Both components of t_1 are longer than $5n + 1$. As in the proof of Proposition 5.12, we have $t_1 \in \mathbb{P}^* \subseteq \mathbb{P}'^*$. Because, $t_2, t_3 \in \mathbb{P}'^*$, we have $t_1 t_2 t_3 = t \in \mathbb{P}'^*$.

Hence, $\mathbb{R}_i \cap P'^*$ is recognizable for $i \in \{0, \ldots, 4\}$, i.e., $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$ is recognizable. □

Proof of Theorem 2.8(3): By Proposition 5.14 and Theorem 3.6, it is undecidable whether for some alphabet Γ with some letter $e \in \Gamma$ and some recognizable language $\mathbb{P}'^*$, the intersection $\binom{\mathsf{e}\Gamma^*}{\Gamma^*} \cap \mathbb{P}'^*$ is recognizable. As in the proof of Theorem 5.7, we define an injective and connected homomorphism $h : \Gamma^* \times \Gamma^* \to \{a, b\}^* \times \{c, d\}^*$. We define h such that for every letter $x \in \Gamma$ the first component of $h\binom{x}{\lambda}$ starts with an a iff $x = e$. Then, Theorem 2.8(3) is immediate. □

References

[1] I. J. Aalbersberg and E. Welzl. Trace languages defined by recognizable string languages. *R.A.I.R.O. - Informatique Théorique et Applications*, 20:103–119, 1986.

[2] J. Berstel. *Transductions and Context-Free Languages*. B. G. Teubner, Stuttgart, 1979.

[3] A. Bertoni, M. Goldwurm, G. Mauri, and N. Sabadini. Counting techniques for inclusion, equivalence and membership problems. Chapter 5 in [8], pages 131-163.

[4] A. Bertoni, G. Mauri, and N. Sabadini. Unambiguous regular trace languages. In G. Demetrovics et al., editors, *Proceedings of the Coll. on Algebra, Combinatorics and Logic in Computer Science*, volume 42 of *Colloquia Mathematica Soc. J. Bolyai*, pages 113–123. North Holland, Amsterdam, 1985.

[5] P. Cartier and D. Foata. *Problèmes combinatoires de commutation et réarrangements*, volume 85 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1969.

[6] M. Clerbout and M. Latteux. Semi-commutations. *Information and Computation*, 73:59–74, 1987.

[7] V. Diekert and Y. Métivier. Partial commutation and traces. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 3, Beyond Words*, pages 457–534. Springer-Verlag, Berlin, 1997.

[8] V. Diekert and G. Rozenberg, editors. *The Book of Traces*. World Scientific, Singapore, 1995.

[9] C. Duboc. Mixed product and asynchronous automata. *Theoretical Computer Science*, 48:183–199, 1986.

[10] S. Eilenberg. *Automata, Languages, and Machines*, Volume A. Academic Press, New York, 1974.

[11] M. Fliess. Matrices de Hankel. *J. Math. Pures et Appl.*, 53:197–224, 1974.

[12] P. Gastin, E. Ochmański, A. Petit, and B. Rozoy. Decidability of the star problem in $A^* \times \{b\}^*$. *Information Processing Letters*, 44(2):65–71, 1992.

[13] D. Giammarresi and A. Restivo. Recognizable picture languages. *International Journal Pattern Recognition and Artificial Intelligence*, 6(2-3):241–256, 1992. Special Issue on *Parallel Image Processing*. M. Nivat, A. Saoudi, and P.S.P. Wangs (Eds.).

[14] D. Giammarresi and A. Restivo. Two-dimensional languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 3, Beyond Words*, pages 215–267. Springer-Verlag, Berlin, 1997.

[15] S. Ginsburg and E. Spanier. Bounded regular sets. In *Proceedings of the AMS*, volume 17:5, pages 1043–1049, 1966.

[16] S. Ginsburg and E. Spanier. Semigroups, Presburger formulas, and languages. *Pacific Journal of Mathematics*, 16:285–296, 1966.

[17] K. Hashiguchi. Limitedness theorem on finite automata with distance functions. *Journal of Computer and System Sciences*, 24:233–244, 1982.

[18] K. Hashiguchi. Improved limitedness theorems on finite automata with distance functions. *Theoretical Computer Science*, 72:1:27–38, 1990.

[19] D. Kirsten. Some undecidability results related to the star problem in trace monoids. Technical Report ISSN 1430-211X, TUD/FI98/07, Dresden University of Technology, Department of Computer Science, May 1998.

[20] D. Kirsten. A connection between the star problem and the finite power property in trace monoids. In P. van Emde Boas et al., editors, *ICALP'99 Proceedings*, volume 1644 of *Lecture Notes in Computer Science*, pages 473–482. Springer-Verlag, Berlin, 1999.

[21] D. Kirsten. *On Decision Problems of Recognizable Trace Languages*. PhD thesis, Dresden University of Technology, Department of Computer Science, 1999.

[22] D. Kirsten. Some undecidability results related to the star problem in trace monoids. In C. Meinel and S. Tison, editors, *STACS'99 Proceedings*, volume 1563 of *Lecture Notes in Computer Science*, pages 227–236. Springer-Verlag, Berlin, 1999.

[23] D. Kirsten and J. Marcinkowski. Two techniques in the area of the star problem. In P. van Emde Boas et al., editors, *ICALP'99 Proceedings*, volume 1644 of *Lecture Notes in Computer Science*, pages 483–492. Springer-Verlag, Berlin, 1999.

[24] D. Kirsten and G. Richomme. Decidability equivalence between the star problem and the finite power problem in trace monoids. Technical Report ISSN 1430-211X, TUD/FI99/03, Dresden University of Technology, Department of Computer Science and LaRIA Technical Report 99-08, Université de Picardie Jules Verne, Amiens, April 1999 (submitted).

[25] A. Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, 1977.

[26] Y. Métivier. Une condition suffisante de reconnaissabilité dans un monoïde partiellement commutatif. *R.A.I.R.O. - Informatique Théorique et Applications*, 20:121–127, 1986.

[27] Y. Métivier and G. Richomme. On the star operation and the finite power property in free partially commutative monoids. In P. Enjalbert et al., editors, *STACS'94 Proceedings*, volume 775 of *Lecture Notes in Computer Science*, pages 341–352. Springer-Verlag, Berlin, 1994.

[28] Y. Métivier and G. Richomme. New results on the star problem in trace monoids. *Information and Computation*, 119(2):240–251, 1995.

[29] E. Ochmański. Recognizable trace languages. Chapter 6 in [8], pages 167-204.

[30] E. Ochmański. *Regular Trace Languages (in Polish)*. PhD thesis, Warszawa, 1984.

[31] M. O. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3:114–125, 1959.

[32] G. Richomme. Some trace monoids where both the star problem and the finite power property problem are decidable. Internal report 755-93, LaBRI - Université Bordeaux I, 1993.

[33] G. Richomme. Some trace monoids where both the star problem and the finite power property problem are decidable. In I. Privara et al., editors, *MFCS'94 Proceedings*, volume 841 of *Lecture Notes in Computer Science*, pages 577–586. Springer-Verlag, Berlin, 1994.

[34] K. Ruohonen. Reversible machines and Post's correspondence problem for biprefix morphisms. *Journal of Information Processing and Cybernetics (EIK)*, 21(12):579–595, 1985.

[35] J. Sakarovitch. On regular trace languages. *Theoretical Computer Science*, 52:59–75, 1987.

[36] J. Sakarovitch. The "last" decision problem for rational trace languages. In I. Simon, editor, *LATIN'92 Proceedings*, volume 583 of *Lecture Notes in Computer Science*, pages 460–473. Springer-Verlag, Berlin, 1992.

[37] S. Yu. Regular languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar*, pages 41–110. Springer-Verlag, Berlin, 1997.