Resynchronizing Classes of Word Relations

María Emilia Descotte
LaBRI ¹

DELTA Meeting in Paris
March 26th 2018

¹Joint work with D. Figueira and G. Puppis
Synchronized pairs of words (over a fixed alphabet \( A \))

**Synchronizing pairs of words**

A synchronization of \((w_1, w_2)\) is a word over \(2 \times A\) so that the projection on \(A\) of positions labeled \(i\) is exactly \(w_i\) for \(i = 1, 2\).
Synchronized pairs of words (over a fixed alphabet \(A\))

A **synchronization** of \((w_1, w_2)\) is a word over \(2 \times A\) so that the projection on \(A\) of positions labeled \(i\) is exactly \(w_i\) for \(i = 1, 2\).

**Example**

\((1, a)(1, b)(2, a)\) and \((1, a)(2, a)(1, b)\) synchronize \((ab, a)\).
Synchronized pairs of words (over a fixed alphabet $\mathbb{A}$)

**Synchronizing pairs of words**

A **synchronization** of $(w_1, w_2)$ is a word over $2 \times \mathbb{A}$ so that the projection on $\mathbb{A}$ of positions labeled $i$ is exactly $w_i$ for $i = 1, 2$.

**Example**

$(1, a)(1, b)(2, a)$ and $(1, a)(2, a)(1, b)$ synchronize $(ab, a)$.

Every word $w \in (2 \times \mathbb{A})^*$ is a synchronization of a unique pair $(w_1, w_2)$ that we denote $[[w]]$.

$$[[1, a)(1, b)(2, a)]] = [[[1, a)(2, a)(1, b)]]] = (ab, a).$$
Synchronized relations

We lift this notion to languages $L \subseteq (\mathbb{2} \times \mathbb{A})^*$

$$[L] = \{[[w]] \mid w \in L\}$$

Example

$\mathbb{A} = \{a, b\}$, $L = ((1, a)(2, a) \cup (1, a)(2, b) \cup (1, b)(2, a) \cup (1, b)(2, b))^*$,

$$[L] = \{(w_1, w_2) \mid |w_1| = |w_2|\}.$$
C-controlled relations

Restrictions on the shape of the projection over $2$

$\Downarrow$

Infinitely many different classes of relations.
C-controlled relations

Restrictions on the shape of the projection over \( 2 \)

\[ \downarrow \]

Infinitely many different classes of relations.

C-controlled words and languages

**C** \( \subseteq \) **2***- regular

- \( w \in (2 \times A)^* \) is **C-controlled** if its projection over \( 2 \) belongs to **C**.
- \( L \subseteq (2 \times A)^* \) is **C-controlled** if all its words are.
**C-controlled relations**

Restrictions on the shape of the projection over $\mathbb{2}$

\[ \Rightarrow \]

Infinitely many different classes of relations.

### C-controlled words and languages

- $C \subseteq \mathbb{2}^*$ regular
  - $w \in (\mathbb{2} \times A)^*$ is **C-controlled** if its projection over $\mathbb{2}$ belongs to $C$.
  - $L \subseteq (\mathbb{2} \times A)^*$ is **C-controlled** if all its words are.

### Examples

- Every $w \in (\mathbb{2} \times A)^*$ is $\mathbb{2}^*$-controlled,
- $(1, a)(1, b)(2, a)$ is $1^*2^*$-controlled,
- $(1, a)(2, a)(1, b)$ isn’t $1^*2^*$-controlled,
- $L$ (previous slide) is $(12)^*$-controlled.
### C-controlled relations

Restrictions on the shape of the projection over \(2\)

\[\Downarrow\]

Infinitely many different classes of relations.

#### C-controlled words and languages

- \(C \subseteq 2^*\) regular
  - \(w \in (2 \times A)^*\) is **C-controlled** if its projection over \(2\) belongs to \(C\).
  - \(L \subseteq (2 \times A)^*\) is **C-controlled** if all its words are.

#### Examples

- Every \(w \in (2 \times A)^*\) is \(2^*\)-controlled,
- \((1, a)(1, b)(2, a)\) is \(1^*2^*\)-controlled,
- \((1, a)(2, a)(1, b)\) isn’t \(1^*2^*\)-controlled,
- \(L\) (previous slide) is \((12)^*\)-controlled.

#### C-controlled relations

Given a regular language \(C \subseteq 2^*\)

\[\text{Rel}(C) = \{[L] \mid L \text{ is reg. and } C\text{-controlled}\}\]
C-controlled relations

Restrictions on the shape of the projection over $2$

Infinitely many different classes of relations.

C-controlled words and languages

$C \subseteq 2^*$ regular

- $w \in (2 \times A)^*$ is C-controlled if its projection over $2$ belongs to $C$.
- $L \subseteq (2 \times A)^*$ is C-controlled if all its words are.

Examples

Every $w \in (2 \times A)^*$ is $2^*$-controlled, $(1, a)(1, b)(2, a)$ is $1^*2^*$-controlled, $(1, a)(2, a)(1, b)$ isn't $1^*2^*$-controlled, $L$ (previous slide) is $(12)^*$-controlled.

C-controlled relations

Given a regular language $C \subseteq 2^*$

$\text{Rel}(C) = \{[L] \mid L \text{ is reg. and C-controlled}\}$

Examples

$\text{Rel}(1^*2^*) = \text{REC}$, $\text{Rel}((12)^*(1^* \cup 2^*)) = \text{REG}$, $\text{Rel}(2^*) = \text{RAT}$.
# Class Containment Problem

| Input: Two regular languages $C, D \subseteq 2^*$ |
| Output: Is $\text{Rel}(C) \subseteq \text{Rel}(D)$? |

**Examples**

- If $C \subseteq D$, then $\text{Rel}(C) \subseteq \text{Rel}(D)$,
- $\text{Rel}(1^*2^*) \subseteq \text{Rel}((12)^*(1^* \cup 2^*))$,
- $\text{Rel}((12)^*(1^* \cup 2^*)) \not\subseteq \text{Rel}(1^*2^*)$,
- $\text{Rel}(1^*2^*) = \text{Rel}(2^*1^*)$,
- $\text{Rel}((12)^*) = \text{Rel}((21)^*)$. 

Synchronized relations

Class Containment Problem

The proof

Conclusions
Previous work

Decidability and complexity
The problem is decidable for $\text{REL}(D) = \text{REC}, \text{REG}$ or $\text{RAT}$.

Resynchronization
The proof is constructive in terms of the automaton:

$\text{Given a NFA for a } C\text{-controlled language } L\text{, one can effectively construct a NFA for a } D\text{-controlled language } L'\text{ such that } [L] = [L'].$

---

Our contribution

We prove that the Class Containment Problem is decidable for arbitrary $C$ and $D$ and, in case of positive answer, we give an effective method for resynchronizing relations.

Proof idea

**Step 1:** Rewrite $C$ and $D$ as finite unions of *simple languages*.

**Step 2:** Characterization for simple languages.

**Step 3:** Induction on the amount of disjuncts in the unions.
Step 1: Decomposition into simple languages

Concat-star languages

\[ C_1^* u_1 \cdots C_n^* u_n \]

with \( C_1, \ldots, C_n \) regular languages, \( u_1, \ldots, u_n \) words.

A component \( C_i^* \) is **homogeneous** if it is contained in \( 1^* \) or \( 2^* \). Otherwise is **heterogeneous**.

- **heterogeneous** if it contains at least one heterogeneous component, otherwise it is **homogeneous**;
- **smooth** if every homogeneous component is \( 1^{k*} \) or \( 2^{k*} \), for some \( k > 0 \), and there are no consecutive homogeneous components;
- **simple** if it has star-height 1 and it is either homogeneous or smooth heterogeneous.

### Examples

- **homogeneous**
  - s.-h. > 1: \((1*1)^*2^*\)
  - s.-h. = 1: \((1*11)^*2^*\)

- **smooth heterogeneous**
  - s.-h. > 1: \(1^*(1^*2)^*2^*\)
  - s.-h. = 1: \(1^*(1^*2)^*2^*\)

- **non-smooth heterogeneous**
  - s.-h. > 1: \(1^*2^*(1^*2)^*\)
  - s.-h. = 1: \(1^*2^*(12)^*\)

- **non concat-star**
  - \((1^*2)* \cup (12)^*\)
  - \((12)^*1^* \cup (12)^*2^*\)
Step 1: Decomposition into simple languages

Every regular language is a finite union of concat-star languages.
Step 1: Decomposition into simple languages

Every regular language is a finite union of **concat-star** languages.

Every **concat-star** language is *Rel-equivalent* to a finite union of **concat-star** languages of star-height 1.
Step 1: Decomposition into simple languages

Every regular language is a finite union of **concat-star** languages.

Every **concat-star** language is *Rel-equivalent* to a finite union of **concat-star** languages of star-height 1.

Every **concat-star** language of star-height 1 is *Rel-equivalent* to a finite union of simple languages.
Step 1: Decomposition into simple languages

Every regular language is a finite union of \textit{concat-star} languages.

Every \textit{concat-star} language is \textit{Rel-equivalent} to a finite union of \textit{concat-star} languages of star-height 1.

Every \textit{concat-star} language of star-height 1 is \textit{Rel-equivalent} to a finite union of simple languages.

Example

\[ \text{REL}((12)^*1^*2^*) = \text{REL}((12)^*1^* \cup (12)^*2^*). \]
Step 2: Characterization for simple languages

Parikh ratio

\[ w \in \mathbb{2}^* \setminus \{\varepsilon\}, \quad \rho(w) = \frac{|w|_1}{|w|}. \]

\[ C \subseteq \mathbb{2}^*, \quad \rho(C) = \{\rho(w) \mid w \in C \setminus \{\varepsilon\}\} \subseteq [0, 1]_\mathbb{Q}. \]
Step 2: Characterization for simple languages

Parikh ratio

\[ w \in 2^* \setminus \{\varepsilon\}, \quad \rho(w) = \frac{|w|_1}{|w|}. \]
\[ C \subseteq 2^*, \quad \rho(C) = \{\rho(w) \mid w \in C \setminus \{\varepsilon\}\} \subseteq [0, 1]_\mathbb{Q}. \]

Synchronizing morphisms

\[ C = C_1^* u_1 \cdots C_n^* u_n, \quad D = D_1^* v_1 \cdots D_m^* v_m. \quad C \xrightarrow{s.m.} D \text{ is} \]
\[ f : [1, \ldots, n] \to [1, \ldots, m] \text{ s.t.} \]

i) \quad \( f \) is monotonic and

ii) \quad \( \rho(C_i^*) \subseteq \rho(D_{f(i)}^*) \) for all \( i = 1, \ldots, n. \)

If \( C \) is homogeneous, we have a s.m. to any \( D \) by convention.
Step 2: Characterization for simple languages

Parikh ratio

\[ w \in \mathcal{2}^* \setminus \{\varepsilon\}, \quad \rho(w) = \frac{|w|_1}{|w|}. \]
\[ C \subseteq \mathcal{2}^*, \quad \rho(C) = \{\rho(w) \mid w \in C \setminus \{\varepsilon\}\} \subseteq [0, 1]_\mathbb{Q}. \]

Synchronizing morphisms

\[ C = C_1^* u_1 \cdots C_n^* u_n, \quad D = D_1^* v_1 \cdots D_m^* v_m. \]
\[ C \xrightarrow{s.m.} D \] is
\[ f : [1, \ldots, n] \to [1, \ldots, m] \text{ s.t.} \]

i) \quad \text{\(f\) is monotonic and}

ii) \quad \rho(C_i^*) \subseteq \rho(D_f^*(i)) \text{ for all } i = 1, \ldots, n.

If \(C\) is homogeneous, we have a s.m. to any \(D\) by convention.
Step 2: Characterization for simple languages

**Proposition**

For all simple languages $C, D \subseteq 2^*$, $\text{Rel}(C') \subseteq \text{Rel}(D)$ iff $\pi(C') \subseteq \pi(D)$ and $C \xrightarrow{s.m.} D$. 

---

**Examples**

- $\text{Rel}((12)^* (112)^*) \subseteq \text{Rel}((12 \cup 11122)^* (121)^* 1^* 2^*)$.
- $\text{Rel}((112)^* (12)^*) \not\subseteq \text{Rel}((12 \cup 11122)^* (121)^* 1^* 2^*)$. 

---

**Conclusions**
Proposition

For all simple languages $C, D \subseteq 2^*$, $\text{Rel}(C) \subseteq \text{Rel}(D)$ iff $\pi(C) \subseteq \pi(D)$ and $C \xrightarrow{s.m.} D$.

Examples

$\text{Rel}((12)^*(112)^*) \subseteq \text{Rel}((12 \cup 11122)^*(121)^*1^*2^*)$,
$\text{Rel}((112)^*(12)^*) \nsubseteq \text{Rel}((12 \cup 11122)^*(121)^*1^*2^*)$. 
### Step 3: Dealing with unions

**Unions on the left**

\[
\text{REL}(C_1 \cup C_2) \subseteq \text{REL}(D) \text{ iff } \text{REL}(C_1) \subseteq \text{REL}(D) \text{ and } \text{REL}(C_2) \subseteq \text{REL}(D).
\]

**Main theorem**

For \( C \) simple and \( D = \bigcup_j D_j \) a finite union of simple languages, the following are equivalent:

i) \( \text{REL}(C) \subseteq \text{REL}(D) \),

ii) \( \pi(C) \subseteq \pi(D) \), \( \exists j \) with \( C \xrightarrow{s.m.} D_j \) and in addition, if \( C \) is heterogeneous, then \( \text{REL}(C \setminus [D_j]_\pi) \subseteq \text{REL}(\bigcup_{j' \neq j} D_{j'}) \).
Our proof gives an effective algorithm to resynchronize relations. We would like to determine the exact complexity.

Other natural questions about this framework: existence and computability of canonical control languages, for which control languages $C \text{ REL}(C)$ is closed under intersection, etc.
Thanks for your attention!