α-β-Factorization and the Binary Case of Simon's Congruence

Pamela Fleischmann[1] Jonas Höfer[2] Annika Huch[1] Dirk Nowotka[1]

[1]Department of Computer Science
Kiel University

[2]Department of Computer Science and Engineering
Gothenburg University

FCT 2023, Trier, Germany

Scattered Factors and Universality

Scattered Factor ($v \preceq w$) v is a *scattered factor* of w if v can be *embedded into* w

- Example

$$\texttt{aba} \preceq \texttt{aabbca} \qquad \texttt{acb} \not\preceq \texttt{aabbca}$$

Scattered Factors and Universality

Scattered Factor ($v \preceq w$) v is a *scattered factor* of w if v can be *embedded into* w

- Example

$$\texttt{aba} \preceq \texttt{aabbca} \qquad \texttt{acb} \not\preceq \texttt{aabbca}$$

Scattered Factors and Universality

Scattered Factor ($v \preceq w$) v is a *scattered factor* of w if v can be *embedded into* w

- Example

$$\texttt{aba} \preceq \texttt{aabbca} \qquad \texttt{acb} \not\preceq \texttt{aabbca}$$

- Partial order compatible with concatenation: $u \preceq v$ and $x \preceq y$ imply $ux \preceq vy$

Scattered Factors and Universality

Scattered Factor ($v \preceq w$) v is a *scattered factor* of w if v can be *embedded into* w

- Example

$$\texttt{aba} \preceq \texttt{aabbca} \qquad \texttt{acb} \not\preceq \texttt{aabbca}$$

- Partial order compatible with concatenation: $u \preceq v$ and $x \preceq y$ imply $ux \preceq vy$
- If $v_1 v_2 \preceq u$ then $v_1 \preceq u_1$ and $v_2 \preceq u_2$ for some factorization $u = u_1 u_2$

Simon's Congruence

Simon's Congruence ($u \sim_k v$) u, v are k-congruent if $\forall w \in \Sigma^{\leq k}. w \preceq u \iff w \preceq v$

- $u \sim_k v$ if u, v have exactly the same scattered factors of length up to k
- Example: $\texttt{abba} \sim_2 \texttt{baba}$, $\texttt{abaaba} \sim_3 \texttt{ababa}$ but $\texttt{aabb} \not\sim_2 \texttt{abba}$

Simon's Congruence

Simon's Congruence ($u \sim_k v$) u, v are k-congruent if $\forall w \in \Sigma^{\leq k}. w \preceq u \iff w \preceq v$

- $u \sim_k v$ if u, v have exactly the same scattered factors of length up to k
- Example: $\texttt{abba} \sim_2 \texttt{baba}$, $\texttt{abaaba} \sim_3 \texttt{ababa}$ but $\texttt{aabb} \not\sim_2 \texttt{abba}$
- Congruence Relation: $u \sim_k v$ and $x \sim_k y$ imply $ux \sim_k vy$
- Finite index because $\Sigma^*/{\sim_k}$ isomorphic to subset of $\mathcal{P}(\Sigma^{\leq k})$

Related Work

Automata and Formal Languages $\Sigma^*/{\sim_k}$ recognizes k-piecewise testable languages

- Piecewise Testable Languages: Boolean combination of $L_v := \{w \in \Sigma^* \mid v \preceq w\}$
- Equivalently, unions of congruence classes of $\sim_k$ for some k
- Simon's Theorem [Sim75]

Combinatorics on Words

- Combinatorial Structure of Classes [Sim72; Lot97; KKS15]
- Reconstruction problems [Lot97; Fle$^+$21]
- Proofs for automata theory results [Klí11]

Algorithmic Problems

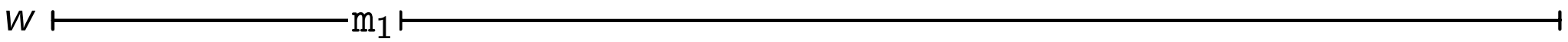
- SimK and MaxSimK [Héb91; FK18; Gaw$^+$21]
- pattern matching problems w.r.t $\sim_k$ [KKH22; Fle$^+$23]

α-β-Factorization

w

Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \text{ScatFact}(w)$

α-β-Factorization



Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathrm{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ

α-β-Factorization

Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

α-β-Factorization

Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

α-β-Factorization

Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

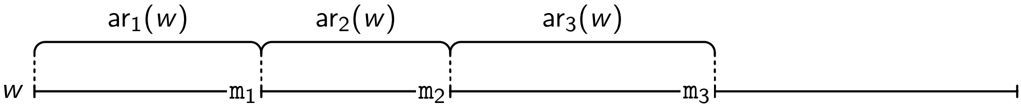
α-β-Factorization

Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

α-β-Factorization

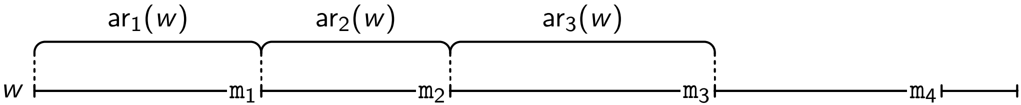


Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

α-β-Factorization

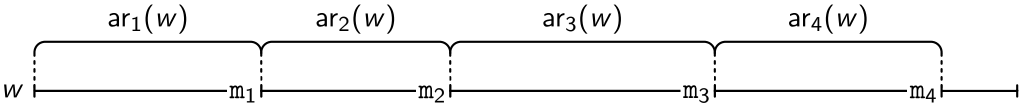


Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$

α-β-Factorization

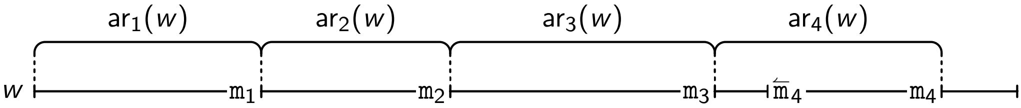


Universality ($\iota(w)$) Maximum $n \in \mathbb{N}$ s.t. $\Sigma^{\leq n} \subseteq \mathsf{ScatFact}(w)$

Arch Factorization [Héb91]

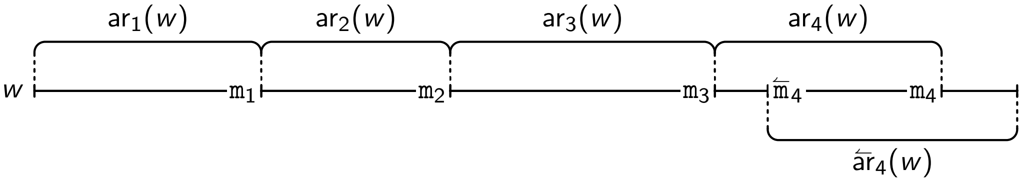
- Repeatedly take shortest prefixes containing Σ
- Last letter $\mathtt{m}_1$ unique in prefix $\implies$ if $\mathtt{m}_1 \cdot v \preceq w$ then $v \preceq \mathsf{ar}_1^{-1}(w) \cdot w$
- $\#$ Arches is $\iota(w)$ because some $\mathtt{m}_1\mathtt{m}_2 \cdots \mathtt{m}_{\#\,\mathsf{Arches}}\mathtt{x}$ does not occur

α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

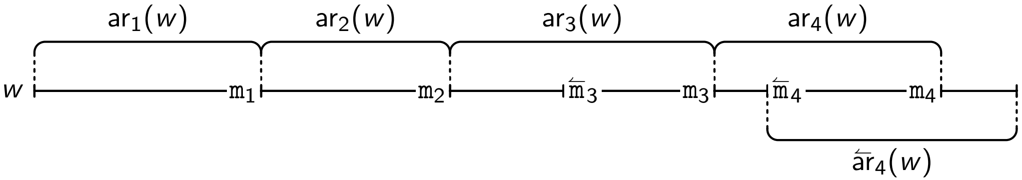
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

– Arches and reverse arches always interleave

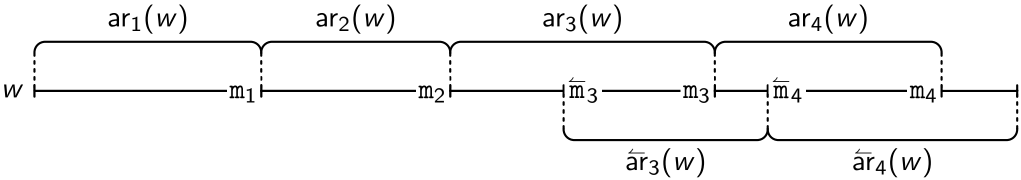
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

– Arches and reverse arches always interleave

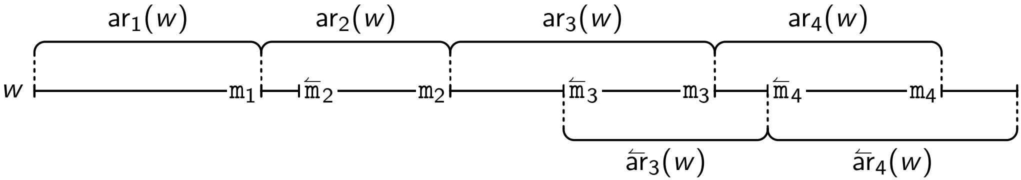
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

– Arches and reverse arches always interleave

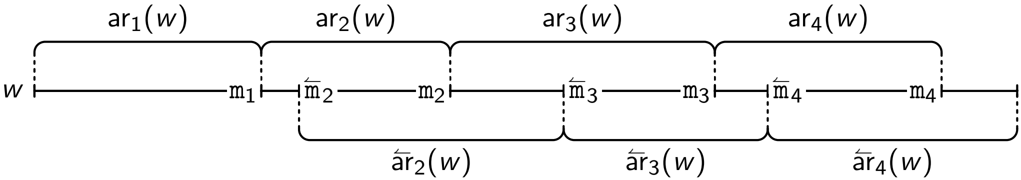
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave

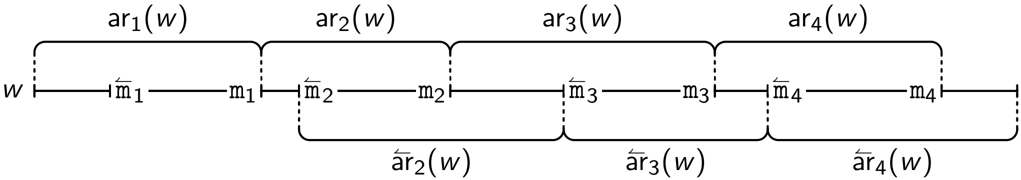
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave

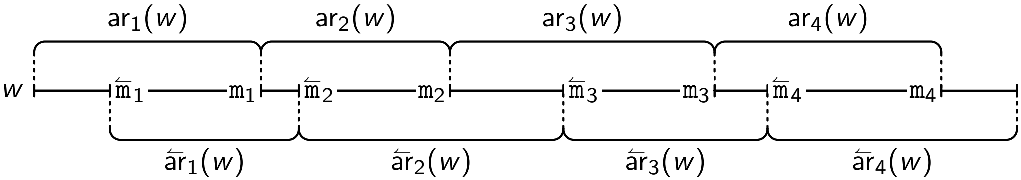
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave

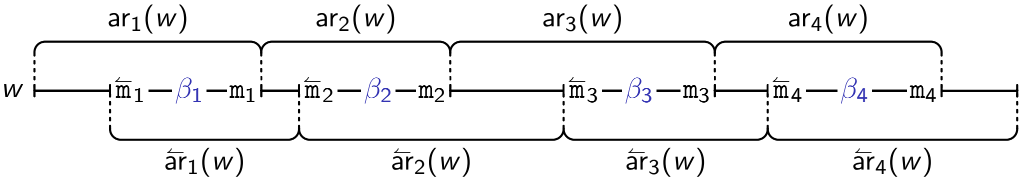
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

– Arches and reverse arches always interleave

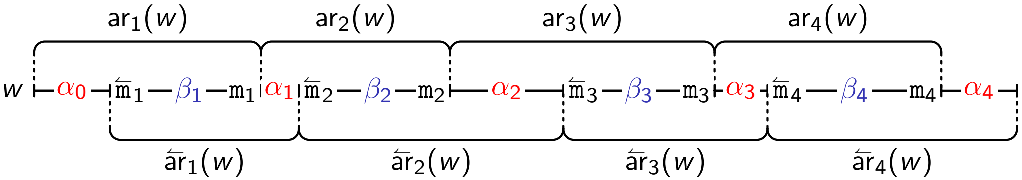
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave
- First and last letter of β_i unique in β_i
- $\beta_i = \mathrm{m}_i = \overleftarrow{\mathrm{m}}_i$ or $\beta_i = \overleftarrow{\mathrm{m}}_i\,\mathrm{core}_i\,\mathrm{m}_i$ with $|\mathrm{alph}(\mathrm{core}_i)| \leq |\Sigma| - 2$

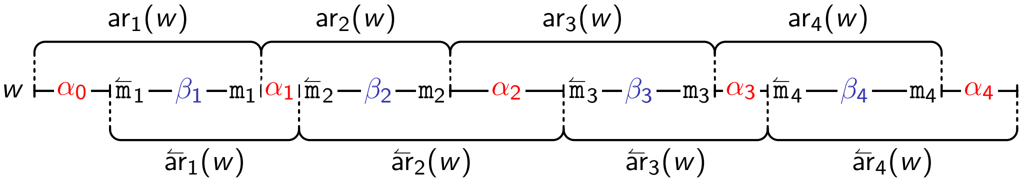
α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave
- First and last letter of β_i unique in β_i
- $\beta_i = \mathsf{m}_i = \overleftarrow{\mathsf{m}}_i$ or $\beta_i = \overleftarrow{\mathsf{m}}_i\,\mathsf{core}_i\,\mathsf{m}_i$ with $|\,\mathsf{alph}(\mathsf{core}_i)| \leq |\Sigma| - 2$
- $|\,\mathsf{alph}(\alpha)| < |\Sigma|$

α-β-Factorization



α-β-Factorization [Fle+22] Arch fact. in both directions

- Arches and reverse arches always interleave
- First and last letter of β_i unique in β_i
- $\beta_i = \mathrm{m}_i = \overleftarrow{\mathrm{m}}_i$ or $\beta_i = \overleftarrow{\mathrm{m}}_i\,\mathrm{core}_i\,\mathrm{m}_i$ with $|\,\mathrm{alph}(\mathrm{core}_i)| \leq |\Sigma| - 2$
- $|\,\mathrm{alph}(\alpha)| < |\Sigma|$

Key Idea Study properties of these factors in general

- In the following always assume $\iota(w) < k$

Lemma

$w \sim_k \tilde{w}$ implies $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ for all $0 \leq i \leq m$ where $m := \iota(w) = \iota(\tilde{w}) < k$.

α-β-Factorization

2 Key Properties

- In the following always assume $\iota(w) < k$

Lemma

$w \sim_k \tilde{w}$ implies $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ for all $0 \leq i \leq m$ where $m := \iota(w) = \iota(\tilde{w}) < k$.
More generally, it implies $\alpha_i \beta_i \cdots \alpha_j \sim_{k-m+(j-i)} \tilde{\alpha}_i \tilde{\beta}_i \cdots \tilde{\alpha}_j$ for all $0 \leq i \leq j \leq m$.

α-β-Factorization

2 Key Properties

- In the following always assume $\iota(w) < k$

Lemma

$w \sim_k \tilde{w}$ implies $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ for all $0 \leq i \leq m$ where $m := \iota(w) = \iota(\tilde{w}) < k$.
More generally, it implies $\alpha_i\beta_i \cdots \alpha_j \sim_{k-m+(j-i)} \tilde{\alpha}_i\tilde{\beta}_i \cdots \tilde{\alpha}_j$ for all $0 \leq i \leq j \leq m$.

- The converse of the second statement holds as well, for that:

α-β-Factorization

2 Key Properties

- In the following always assume $\iota(w) < k$

Lemma

*$w \sim_k \tilde{w}$ implies $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ for all $0 \leq i \leq m$ where $m := \iota(w) = \iota(\tilde{w}) < k$.
More generally, it implies $\alpha_i\beta_i \cdots \alpha_j \sim_{k-m+(j-i)} \tilde{\alpha}_i\tilde{\beta}_i \cdots \tilde{\alpha}_j$ for all $0 \leq i \leq j \leq m$.*

- The converse of the second statement holds as well, for that:

Lemma

Up to $\sim_k$, a factor of the form $\alpha_i\beta_i \cdots \alpha_j$ of w can be replaced by a $(k - \iota(w) + (j - 1))$-equivalent one.

- Freely exchange α bordered factors w.r.t $\sim_{k-\ell}$ where ℓ is # β factors outside

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ *with* $0 < m := \iota(w) = \iota(\tilde{w}) < k$, *we have*

$$w \sim_k \tilde{w} \iff \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \leq i \leq m.$$

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ with $0 < m := \iota(w) = \iota(\tilde{w}) < k$, we have

$$w \sim_k \tilde{w} \iff \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \leq i \leq m.$$

Proof Sketch. Apply Lemma 2 repeatedly (similar to Karandikar et al. [KKS15])

$$\begin{aligned}
\alpha_0\beta_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots &\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\beta_2\alpha_2\beta_3\alpha_3\cdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots \qquad (\alpha_1 \sim_{k-m} \tilde{\alpha}_1 \text{ by Lemma 1}) \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\beta_3\alpha_3\cdots \\
&\vdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\tilde{\beta}_3\tilde{\alpha}_3\cdots.
\end{aligned}$$

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ with $0 < m := \iota(w) = \iota(\tilde{w}) < k$, we have

$$w \sim_k \tilde{w} \quad \Longleftrightarrow \quad \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \leq i \leq m.$$

Proof Sketch. Apply Lemma 2 repeatedly (similar to Karandikar et al. [KKS15])

$$\begin{aligned}
\alpha_0\beta_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots &\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\beta_2\alpha_2\beta_3\alpha_3\cdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots \qquad (\alpha_1 \sim_{k-m} \tilde{\alpha}_1 \text{ by Lemma 1}) \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\beta_3\alpha_3\cdots \\
&\vdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\tilde{\beta}_3\tilde{\alpha}_3\cdots .
\end{aligned}$$

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ with $0 < m := \iota(w) = \iota(\tilde{w}) < k$, we have

$$w \sim_k \tilde{w} \quad \Longleftrightarrow \quad \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \leq i \leq m.$$

Proof Sketch. Apply Lemma 2 repeatedly (similar to Karandikar et al. [KKS15])

$$\begin{aligned}
\alpha_0\beta_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots &\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\beta_2\alpha_2\beta_3\alpha_3\cdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots \qquad (\alpha_1 \sim_{k-m} \tilde{\alpha}_1 \text{ by Lemma 1}) \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\beta_3\alpha_3\cdots \\
&\vdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\tilde{\beta}_3\tilde{\alpha}_3\cdots .
\end{aligned}$$

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ with $0 < m := \iota(w) = \iota(\tilde{w}) < k$, we have

$$w \sim_k \tilde{w} \quad \Longleftrightarrow \quad \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \le i \le m.$$

Proof Sketch. Apply Lemma 2 repeatedly (similar to Karandikar et al. [KKS15])

$$\begin{aligned}
\alpha_0\beta_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots &\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\beta_2\alpha_2\beta_3\alpha_3\cdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots \qquad (\alpha_1 \sim_{k-m} \tilde{\alpha}_1 \text{ by Lemma 1}) \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\beta_3\alpha_3\cdots \\
&\vdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\tilde{\beta}_3\tilde{\alpha}_3\cdots .
\end{aligned}$$

α-β-Decomposition

Theorem

For $w, \tilde{w} \in \Sigma^*$ with $0 < m := \iota(w) = \iota(\tilde{w}) < k$, we have

$$w \sim_k \tilde{w} \quad \Longleftrightarrow \quad \alpha_{i-1}\beta_i\alpha_i \sim_{k-m+1} \tilde{\alpha}_{i-1}\tilde{\beta}_i\tilde{\alpha}_i \text{ for all } 1 \leq i \leq m.$$

Proof Sketch. Apply Lemma 2 repeatedly (similar to Karandikar et al. [KKS15])

$$\begin{aligned}
\alpha_0\beta_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots &\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\beta_2\alpha_2\beta_3\alpha_3\cdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\alpha_1\beta_2\alpha_2\beta_3\alpha_3\cdots && (\alpha_1 \sim_{k-m} \tilde{\alpha}_1 \text{ by Lemma 1}) \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\beta_3\alpha_3\cdots \\
&\vdots \\
&\sim_k \tilde{\alpha}_0\tilde{\beta}_1\tilde{\alpha}_1\tilde{\beta}_2\tilde{\alpha}_2\tilde{\beta}_3\tilde{\alpha}_3\cdots .
\end{aligned}$$

□

Binary Case

Initial Observations

- We consider $\Sigma_2 := \{\mathtt{a}, \mathtt{b}\}$ and set $\overline{\mathtt{a}} := \mathtt{b}$ and $\overline{\mathtt{b}} := \mathtt{a}$

Lemma

For $w \in \Sigma_2^*$ we have

1. $\beta_i \in \{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}\}$
2. *if $\beta_i = \mathtt{x}$ then $\alpha_{i-1}, \alpha_i \in \overline{\mathtt{x}}^+$*
3. *if $\beta_i = \mathtt{x}\overline{\mathtt{x}}$ then $\alpha_{i-1} \in \mathtt{x}^*$ and $\alpha_i \in \overline{\mathtt{x}}^*$*

Binary Case

Initial Observations

- We consider $\Sigma_2 := \{\mathtt{a}, \mathtt{b}\}$ and set $\overline{\mathtt{a}} := \mathtt{b}$ and $\overline{\mathtt{b}} := \mathtt{a}$

Lemma

For $w \in \Sigma_2^*$ we have

1. $\beta_i \in \{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}\}$
2. *if $\beta_i = \mathtt{x}$ then $\alpha_{i-1}, \alpha_i \in \overline{\mathtt{x}}^+$*
3. *if $\beta_i = \mathtt{x}\overline{\mathtt{x}}$ then $\alpha_{i-1} \in \mathtt{x}^*$ and $\alpha_i \in \overline{\mathtt{x}}^*$*

- We have an even stronger characterization in the binary case!

Lemma

For $w \sim_k \tilde{w}$ with $m := \iota(w) = \iota(\tilde{w}) < k$, we have $\beta_i = \tilde{\beta}_i$ for all $1 \leq i \leq m$.

Binary Case

Some Characterizations

Theorem

For $w, \tilde{w} \in \Sigma_2^*$ *with* $\iota(w), \iota(\tilde{w}) < k$ *we have* $w \sim_k \tilde{w}$ *iff* $m := \iota(w) = \iota(\tilde{w})$, $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ *for all* $0 \leq i \leq m$, *and* $\beta_i = \tilde{\beta}_i$ *for all* $1 \leq i \leq m$.

Binary Case

Some Characterizations

Theorem

For $w, \tilde{w} \in \Sigma_2^*$ with $\iota(w), \iota(\tilde{w}) < k$ we have $w \sim_k \tilde{w}$ iff $m := \iota(w) = \iota(\tilde{w})$, $\alpha_i \sim_{k-m} \tilde{\alpha}_i$ for all $0 \leq i \leq m$, and $\beta_i = \tilde{\beta}_i$ for all $1 \leq i \leq m$.

- By Simon [Sim72; Lot97], we have $|[w]_{\sim_k}| \in \{1, \infty\}$
- We can characterize these in the binary case easily

Theorem

For $w \in \Sigma_2^*$ we have $|[w]_{\sim_k}| = 1$ iff $\iota(w) < k$ and $|\alpha_i| < k - m$.

Enumeration of $|\Sigma_2^*/{\sim_k}|$

Core Idea Separate by $\iota(w)$, then use characterization

- Know all possibilities for α_i, β_j but not all combinations possible

Enumeration of $|\Sigma_2^*/{\sim_k}|$

Core Idea Separate by $\iota(w)$, then use characterization

- Know all possibilities for α_i, β_j but not all combinations possible

Observation Fixed k is problematic

- Given $w = \alpha_0\beta_1 \cdots \alpha_{\iota(w)}$ with α_i representing $\sim_{k-m}$ class:
  - extend to $w' := w\beta\alpha$ for some matching β, α
  - α_i' only fixed up to $\sim_{k-m-1}$

Enumeration of $|\Sigma_2^*/\sim_k|$

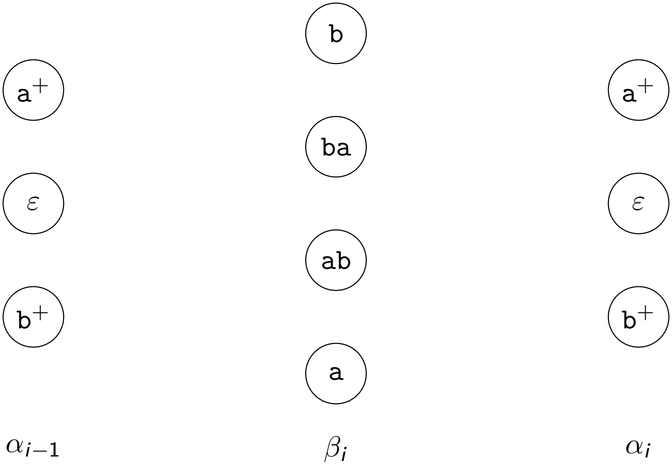
Core Idea Separate by $\iota(w)$, then use characterization

- Know all possibilities for α_i, β_j but not all combinations possible

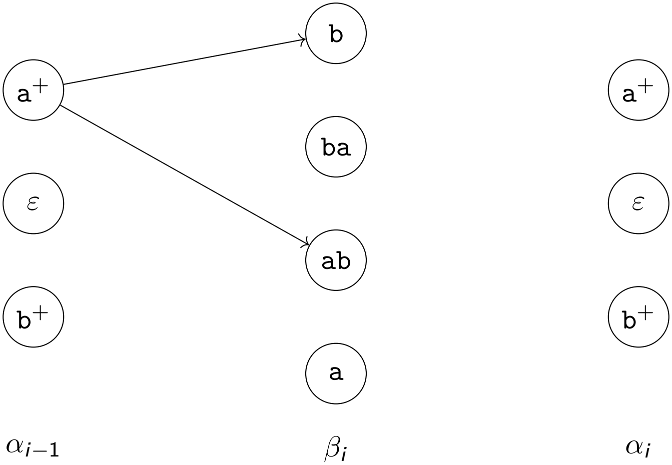
Observation Fixed k is problematic

- Given $w = \alpha_0\beta_1 \cdots \alpha_{\iota(w)}$ with α_i representing $\sim_{k-m}$ class:
  - extend to $w' := w\beta\alpha$ for some matching β, α
  - α_i' only fixed up to $\sim_{k-m-1}$
- Count for all k at once!
  - Increase k and $m = \iota(w)$ together: append $\beta\alpha$, consider result up to $\sim_{k+1}$
  - Fix difference $\Delta := k - m \geq 1$

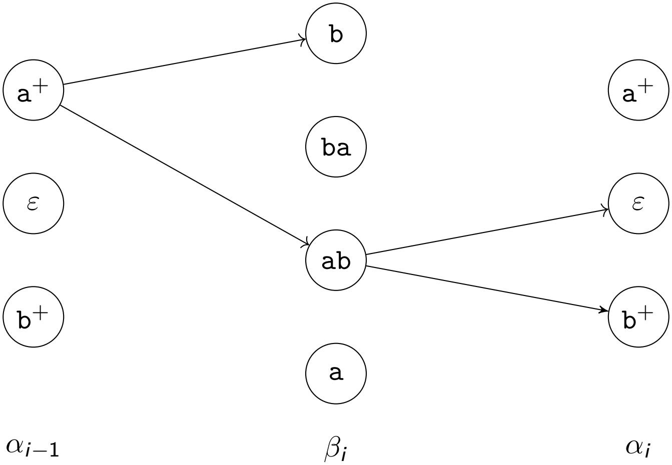
Enumeration of $|\Sigma_2^*/\sim_k|$



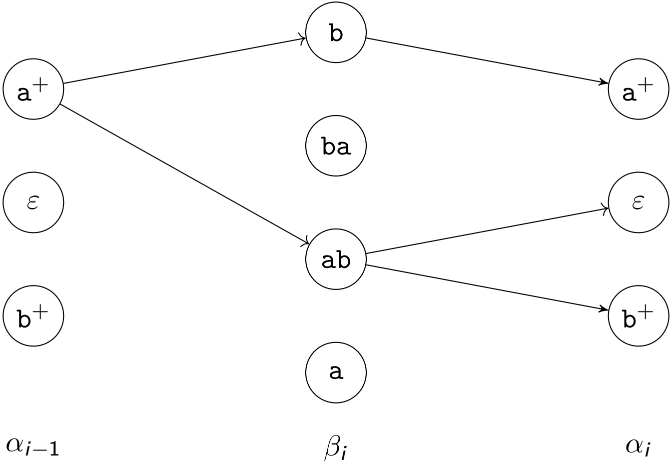
Enumeration of $|\Sigma_2^*/{\sim_k}|$



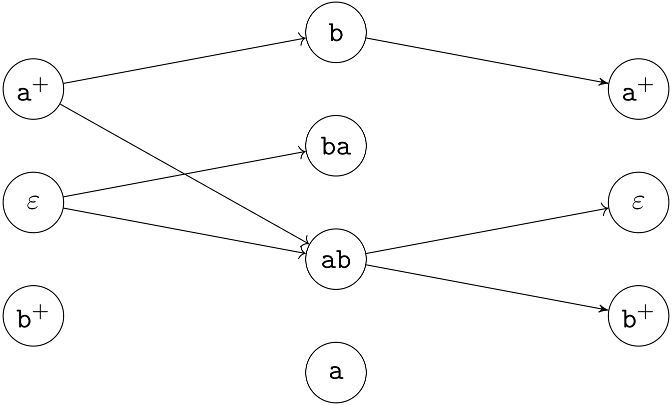
Enumeration of $|\Sigma_2^*/{\sim_k}|$



Enumeration of $|\Sigma_2^*/\sim_k|$

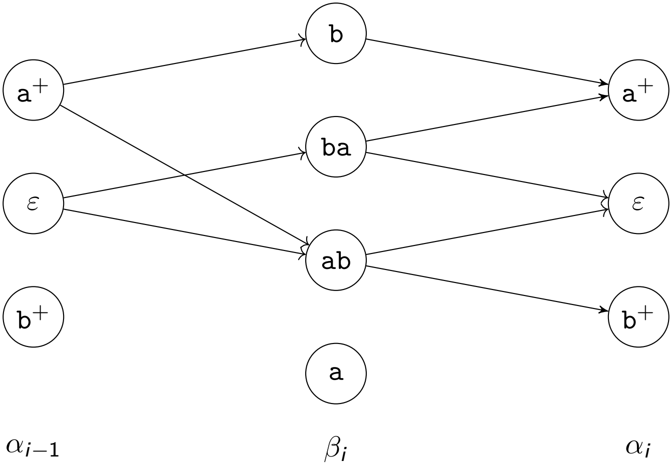


Enumeration of $|\Sigma_2^*/\!\sim_k|$

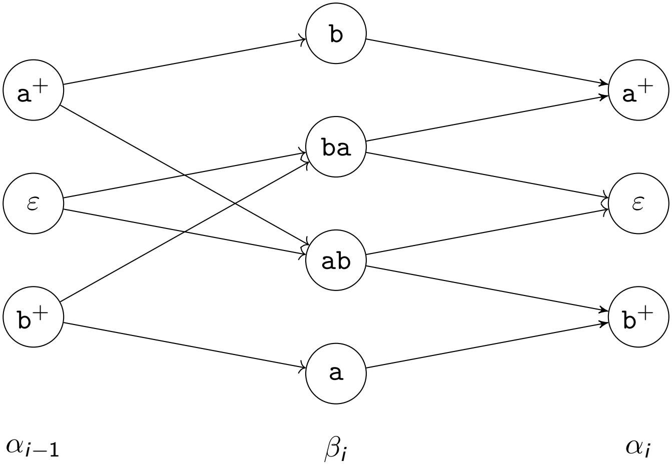


α_{i-1} β_i α_i

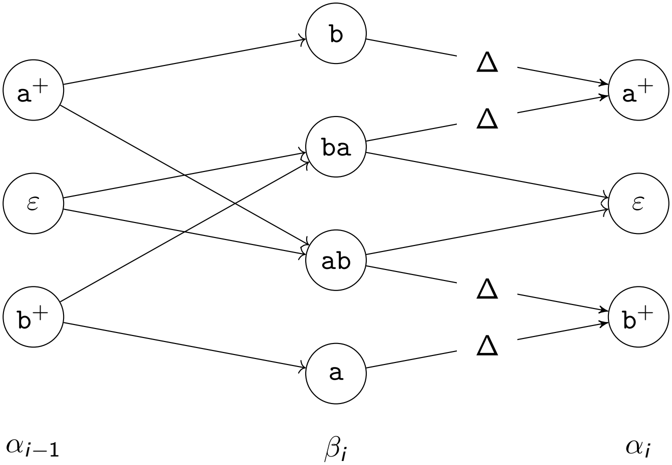
Enumeration of $|\Sigma_2^*/{\sim_k}|$



Enumeration of $|\Sigma_2^*/{\sim_k}|$



Enumeration of $|\Sigma_2^*/\sim_k|$



Enumeration of $|\Sigma_2^*/\sim_k|$

- Number of classes of words with ℓ arches w.r.t. $\ell + \Delta$

$$c_\Delta(\ell) := \left|\{w \in \Sigma_2^* \mid \iota(w) = \ell\}/\sim_{\Delta+\ell}\right| = \left\| \begin{bmatrix} \Delta & \Delta & \Delta \\ 1 & 2 & 1 \\ \Delta & \Delta & \Delta \end{bmatrix}^\ell \cdot \begin{bmatrix} \Delta \\ 1 \\ \Delta \end{bmatrix} \right\|_1$$

Enumeration of $|\Sigma_2^*/\sim_k|$

- Number of classes of words with ℓ arches w.r.t. $\ell + \Delta$

$$c_\Delta(\ell) := \left| \{ w \in \Sigma_2^* \mid \iota(w) = \ell \} / \sim_{\Delta+\ell} \right| = \left\| \begin{bmatrix} \Delta & \Delta & \Delta \\ 1 & 2 & 1 \\ \Delta & \Delta & \Delta \end{bmatrix}^\ell \cdot \begin{bmatrix} \Delta \\ 1 \\ \Delta \end{bmatrix} \right\|_1$$

- Using some linear algebra, we obtain

$$c_\Delta(-1) = 1 \qquad c_\Delta(0) = 2 \cdot \Delta + 1$$
$$c_\Delta(\ell + 2) = 2 \cdot (\Delta + 1) \cdot c_\Delta(\ell + 1) - 2 \cdot \Delta \cdot c_\Delta(\ell).$$

Enumeration of $|\Sigma_2^*/{\sim_k}|$

Theorem

For all $k \in \mathbb{N}$, we have

$$|\Sigma_2^*/{\sim_k}| = 1 + \sum_{\Delta=1}^{k} c_\Delta(k - \Delta).$$

Enumeration of $|\Sigma_2^*/{\sim_k}|$

Theorem

For all $k \in \mathbb{N}$, we have

$$|\Sigma_2^*/{\sim_k}| = 1 + \sum_{\Delta=1}^{k} c_\Delta(k - \Delta).$$

- $s(k)$ depends on values from $s_1, \ldots, s_k$
- We can enumerate $s(k)$ by storing a list of the last two values of c_i

Enumeration of $|\Sigma_2^*/{\sim_k}|$

Theorem

For all $k \in \mathbb{N}$, we have

$$|\Sigma_2^*/{\sim_k}| = 1 + \sum_{\Delta=1}^{k} c_\Delta(k - \Delta).$$

- $s(k)$ depends on values from $s_1, \ldots, s_k$
- We can enumerate $s(k)$ by storing a list of the last two values of c_i

$$1, 4, 16, 68, 312, 1560, 8528, 50864, 329248, 2298592, 17203264, 137289920, 1162805376, 10409679744, 98146601216, 971532333824, 10068845515264, \ldots$$

Towards Σ_3

Modi Always $|\beta_i| = 1$ or $\beta_i = \overleftarrow{\mathtt{m}}_i \, \mathsf{core}_i \, \mathtt{m}_i$ but

Towards Σ_3

Modi Always $|\beta_i| = 1$ or $\beta_i = \overleftarrow{\mathtt{m}}_i\,\mathsf{core}_i\,\mathtt{m}_i$ but

$$\mathtt{abab} \cdot \mathtt{abc} \cdot \mathtt{c} \sim_3 \mathtt{abab} \cdot \mathtt{bac} \cdot \mathtt{c}$$

Towards Σ_3

Modi Always $|\beta_i| = 1$ or $\beta_i = \overleftarrow{\mathtt{m}}_i\,\mathsf{core}_i\,\mathtt{m}_i$ but

$$\mathtt{abab} \cdot \mathtt{abc} \cdot \mathtt{c} \sim_3 \mathtt{abab} \cdot \mathtt{bac} \cdot \mathtt{c}$$

Core Always $|\mathsf{alph}(\mathsf{core}_i)| \leq 1$ but

Towards Σ_3

Modi Always $|\beta_i| = 1$ or $\beta_i = \overleftarrow{\mathtt{m}}_i\, \mathsf{core}_i\, \mathtt{m}_i$ but

$$\mathtt{abab} \cdot \mathtt{abc} \cdot \mathtt{c} \sim_3 \mathtt{abab} \cdot \mathtt{bac} \cdot \mathtt{c}$$

Core Always $|\mathsf{alph}(\mathsf{core}_i)| \leq 1$ but

$$\mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-2}\mathtt{c} \cdot \varepsilon \not\sim_k \mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-1}\mathtt{c} \cdot \varepsilon$$

$$\mathtt{b}^n\mathtt{a}\color{red}{\mathtt{b}} \cdot \mathtt{ab}^{k-2}\mathtt{c} \cdot \varepsilon \sim_k \mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-1}\mathtt{c} \cdot \varepsilon$$

Towards Σ_3

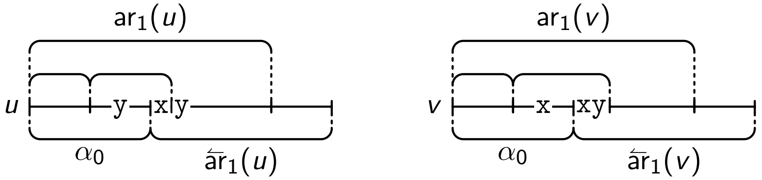
Modi Always $|\beta_i| = 1$ or $\beta_i = \overleftarrow{\mathtt{m}}_i\, \mathrm{core}_i\, \mathtt{m}_i$ but

$$\mathtt{abab} \cdot \mathtt{abc} \cdot \mathtt{c} \sim_3 \mathtt{abab} \cdot \mathtt{bac} \cdot \mathtt{c}$$

Core Always $|\mathrm{alph}(\mathrm{core}_i)| \leq 1$ but

$$\mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-2}\mathtt{c} \cdot \varepsilon \not\sim_k \mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-1}\mathtt{c} \cdot \varepsilon$$

$$\mathtt{b}^n\mathtt{ab} \cdot \mathtt{ab}^{k-2}\mathtt{c} \cdot \varepsilon \sim_k \mathtt{b}^n\mathtt{a} \cdot \mathtt{ab}^{k-1}\mathtt{c} \cdot \varepsilon$$



Card	$\mathsf{alph}(\alpha_0)$	$\mathsf{alph}(\alpha_1)$	β RegExp
2–2	$\{\mathtt{a},\mathtt{b}\}$	$\{\mathtt{a},\mathtt{c}\}$	$\mathtt{ba}^*\mathtt{c}$
	$\{\mathtt{a},\mathtt{b}\}$	$\{\mathtt{a},\mathtt{b}\}$	$\mathtt{c}$
2–1	$\{\mathtt{a},\mathtt{b}\}$	$\{\mathtt{c}\}$	$(\mathtt{a}^+\mathtt{b} \mid \mathtt{b}^+\mathtt{a})\mathtt{c}$
	$\{\mathtt{a},\mathtt{b}\}$	$\{\mathtt{a}\}$	$\mathtt{ba}^*\mathtt{c}$
2–0	$\{\mathtt{a},\mathtt{b}\}$	$\emptyset$	$(\mathtt{a}^+\mathtt{b} \mid \mathtt{b}^+\mathtt{a})\mathtt{c}$
1–1	$\{\mathtt{a}\}$	$\{\mathtt{b}\}$	$\mathtt{ab}^+\mathtt{c} \mid \mathtt{ac}^+\mathtt{b} \mid \mathtt{ca}^+\mathtt{b}$
	$\{\mathtt{a}\}$	$\{\mathtt{a}\}$	$\mathtt{ba}^*\mathtt{c}$
1–0	$\{\mathtt{a}\}$	$\emptyset$	$\mathtt{ba}^*\mathtt{c} \mid \mathtt{ab}^+\mathtt{c}$
0–0	$\emptyset$	$\emptyset$	$\mathtt{ab}^+\mathtt{c}$

- $\mathsf{a},\mathsf{b},\mathsf{c} \in \Sigma_3$ different; permutation of modi occurs only in case 3 and 5
- length of core$_i$ determined by $\iota(\alpha_i), \mathsf{alph}(\alpha_i), \mathsf{re}(\alpha_0), \overleftarrow{\mathsf{re}}(\alpha_1)$ (depends on k)

Questions?

Bibliography I

[FK18] Lukas Fleischer and Manfred Kufleitner. “Testing Simon’s Congruence”. In: *43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018* (Liverpool, UK, Aug. 27–31, 2018). Ed. by Igor Potapov, Paul G. Spirakis, and James Worrell. Vol. 117. Leibniz International Proceedings in Informatics. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018, 62:1–62:13. URL: https://doi.org/10.4230/LIPIcs.MFCS.2018.62.

[Fle+21] Pamela Fleischmann et al. “Reconstructing Words from Right-Bounded-Block Words”. In: *Int. J. Found. Comput. Sci.* 32.6 (2021), pp. 619–640. URL: https://doi.org/10.1142/S0129054121420016.

Bibliography II

[Fle+22] Pamela Fleischmann et al. “Nearly k-Universal Words – Investigating a Part of Simon’s Congruence”. In: *Descriptional Complexity of Formal Systems – 24th IFIP WG 1.02 International Conference, DCFS 2022, Proceedings* (Debrecen, Hungary, Aug. 29–31, 2022). Ed. by Yo-Sub Han and György Vaszil. Vol. 13439. Lecture Notes in Computer Science. Springer, 2022, pp. 57–71. URL: https://doi.org/10.1007/978-3-031-13257-5%5C_5.

[Fle+23] Pamela Fleischmann et al. “Matching Patterns with Variables Under Simon’s Congruence”. In: *CoRR* abs/2308.08374 (2023). arXiv: 2308.08374 [cs.FL]. URL: https://doi.org/10.48550/arXiv.2308.08374.

Bibliography III

[Gaw+21] Pawel Gawrychowski et al. “Efficiently Testing Simon’s Congruence”. In: *38th International Symposium on Theoretical Aspects of Computer Science, STACS 2021* (Saarbrücken, Germany (Virtual Conference), Mar. 16–19, 2021). Ed. by Markus Bläser and Benjamin Monmege. Vol. 187. Leibniz International Proceedings in Informatics. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, 34:1–34:18. URL: https://doi.org/10.4230/LIPIcs.STACS.2021.34.

[Héb91] Jean-Jacques Hébrard. “An Algorithm for Distinguishing Efficiently Bit-Strings by their Subsequences”. In: *Theoretical Computer Science* 82.1 (1991), pp. 35–49. URL: https://doi.org/10.1016/0304-3975(91)90170-7.

Bibliography IV

[KKH22] Sungmin Kim, Sang-Ki Ko, and Yo-Sub Han. "Simon's Congruence Pattern Matching". In: *33rd International Symposium on Algorithms and Computation, ISAAC 2022* (Seoul, Korea, Dec. 19–21, 2022). Ed. by Sang Won Bae and Heejin Park. Vol. 248. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 60:1–60:17. URL: https://doi.org/10.4230/LIPIcs.ISAAC.2022.60.

[KKS15] Prateek Karandikar, Manfred Kufleitner, and Philippe Schnoebelen. "On the Index of Simon's Congruence for Piecewise Testability". In: *Information Processing Letters* 115.4 (2015), pp. 515–519. URL: https://doi.org/10.1016/j.ipl.2014.11.008.

[Klí11] Ondrej Klíma. "Piecewise testable languages via combinatorics on words". In: *Discrete Mathematics* 311.20 (2011), pp. 2124–2127. URL: https://doi.org/10.1016/j.disc.2011.06.013.

Bibliography V

[Lot97] M. Lothaire. *Combinatorics on Words*. 2nd ed. Cambridge Mathematical Library. Cambridge University Press, 1997. ISBN: 978-0-521-59924-5. URL: https://doi.org/10.1017/CBO9780511566097.

[Sim72] Imre Simon. "Hierarchies of Events with Dot-depth One". PhD thesis. University of Waterloo, Department of Applied Analysis and Computer Science, 1972.

[Sim75] Imre Simon. "Piecewise Testable Events". In: *Automata Theory and Formal Languages, 2nd GI Conference* (Kaiserslautern, Germany, May 20–23, 1975). Ed. by H. Barkhage. Vol. 33. Lecture Notes in Computer Science. Springer, 1975, pp. 214–222. URL: https://doi.org/10.1007/3-540-07407-4_23.