On The Reversibility of Parallel Insertion, and Its Relation to Comma Codes

Bo Cui, Lila Kari, Shinnosuke Seki
Department of Computer Science
University of Western Ontario

Introduction

- Operations are essential in generating words and languages.
  - Catenation: $a \cdot b = ab$
  - Right quotient: $abb/bb = a$
  - Kleene Star: $a^* = \{\lambda, a, aa, aa \cdots\}$
  - Sequential insertion and deletion (Kari, L. 1991)
  - Parallel insertion and deletion (Kari, L. 1991)
- The reversibility of parallel insertion and deletion
  i.e. $(L_1 \Leftarrow L_2) \Rightarrow L_2 = L_1$
- A complete solution to $(w \Leftarrow u) \Rightarrow u = \{w\}$
- Comma codes and their properties
- A generalization of comma codes: comma intercodes

Definition of Parallel Insertion (Kari, L. 1991)

For two languages $L_1, L_2 \subseteq \Sigma^*$, the parallel insertion of L_2 into L_1 generates

$$L_1 \Leftarrow L_2 = \bigcup_{n \geq 0,\ a_1, \ldots, a_n \in \Sigma \text{ s.t. } a_1 a_2 \cdots a_n \in L_1} L_2 a_1 L_2 a_2 \cdots L_2 a_n L_2.$$

Example

For $L_1 = \{cd\}$ and $L_2 = \{a, b\}$,

$$\begin{aligned} & L_1 \Leftarrow L_2 \\ = \ & L_2 c L_2 d L_2 \\ = \ & \{acada, acadb, acbda, acbdb, bcada, bcadb, bcbda, bcbdb\}. \end{aligned}$$

Definition of Parallel Deletion (Kari, L. 1991)

For two languages $L_1, L_2 \subseteq \Sigma^*$, the parallel deletion of L_2 from L_1 is defined as $L_1 \Rightarrow L_2 = \bigcup_{u \in L_1} (u \Rightarrow L_2)$,
where

$$\begin{aligned} u \Rightarrow L_2 &= \big\{ u_1 u_2 \cdots u_k u_{k+1} \mid u_1, \ldots, u_{k+1} \in \Sigma^*,\ k \geq 1, \\ &\quad u \in u_1 L_2 u_2 L_2 \cdots L_2 u_{k+1} \text{ and } \mathrm{F}(u_i) \cap (L_2 \setminus \{\lambda\}) = \emptyset \\ &\quad \text{for all } 1 \leq i \leq k+1 \big\}. \end{aligned}$$

Example

Let $L_1 = \{ababab a, aababa, abaabaaba\}$ and $L_2 = \{aba\}$. Then

$$\begin{aligned} & L_1 \Rightarrow L_2 \\ = \;& (\{abababa\} \Rightarrow L_2) \cup (\{aababa\} \Rightarrow L_2) \cup (\{abaabaaba\} \Rightarrow L_2) \\ = \;& \{b, abba\} \cup \{aba, aab\} \cup \{\lambda\} = \{b, abba, aba, aab, \lambda\}. \end{aligned}$$

Objective

Question

$$\textit{When does } (L_1 \Leftarrow L_2) \Rightarrow L_2 = L_1 \textit{?}$$

Example

Let $L_1 = \{b\}$ and $L_2 = \{aba\}$, then

$$\begin{aligned} L_1 \Leftarrow L_2 &= \{abababa\} \\ (L_1 \Leftarrow L_2) \Rightarrow L_2 &= \{abababa\} \Rightarrow \{aba\} \\ &= \{b, abba\} \end{aligned}$$

Preliminaries

Primitive word: if $u = v^n$ implies $n = 1$ and $u = v$ for any $v \in \Sigma^+$.

- e.g. $abababab \notin Q$, $aaba \in Q$.

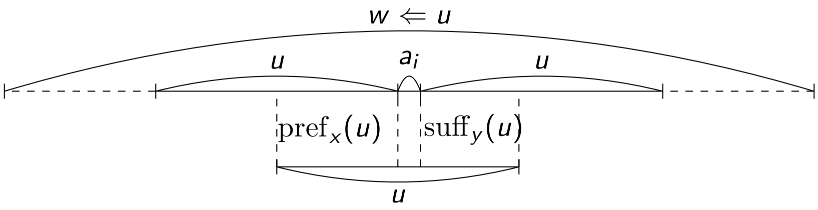
Primitive root: any non-primitive word can be written as a power of a unique primitive word.

If nonempty words $x, y, z \in \Sigma^+$ satisfy $xy = yz$, then there exist $\alpha, \beta \in \Sigma^*$ such that $\alpha\beta$ is primitive, $x = (\alpha\beta)^i$, $y = (\alpha\beta)^j\alpha$, and $z = (\beta\alpha)^i$ for some $i \geq 1$ and $j \geq 0$.

Bordered word: $w = xy = yz$ for some $x, y, z \in \Sigma^+$. Clearly, $w = (\alpha\beta)^k\alpha$ such that $\alpha\beta \in Q$ and $k \geq 1$.

Bordered primitive word: primitive and bordered, e.g. *ababa*.

Conditions for $(w \Leftarrow u) \Rightarrow u = \{w\}$?



Definition

$$X \;=\; \left\{u \in \Sigma^+ \mid \mathrm{pref}_x(u) \neq \mathrm{suff}_x(u) \textit{ or } \mathrm{pref}_y(u) \neq \mathrm{suff}_y(u) \right.$$
$$\left. \textit{for any } (x, y) \in N^2 \textit{ with } x + y + 1 = |u| \right\}.$$

Proposition

If $u \in X$, *then* $(w \Leftarrow u) \Rightarrow u = \{w\}$ *for any* $w \in \Sigma^*$.

Characterization of X

Σ^+	Q		$\overline{Q}$
X	$U^{>1}$	$Q_B^{(>1)}$	$N_{(>1)}$
$\overline{X}$	Σ	$Q_B^{(=1)}$	$\{aa^+ \mid a \in \Sigma\}$

$N_{(>1)}$ denotes the set of non-primitive words whose primitive root is longer than 1

$Q_B^{(=1)}$ denotes the set of all bordered primitive words w that can be written as $(\alpha\beta)^k\alpha$ with $|\beta| = 1$. e.g. $aabaabaa \in Q_B^{(=1)}$.

$Q_B^{(>1)} = Q_B \setminus Q_B^{(=1)}$. e.g. $aaabaa, abbabba \in Q_B^{(>1)}$.

Solution to $(w \Leftarrow u) \Rightarrow u = \{w\}$

Proposition

If $u \in X$, then $(w \Leftarrow u) \Rightarrow u = \{w\}$ for any $w \in \Sigma^*$.

Proposition

Let $w \in \Sigma^*$ and $u = a^k$ for some $a \in \Sigma$ and $k \geq 1$. Then $(w \Leftarrow u) \Rightarrow u = \{w\}$ if and only if

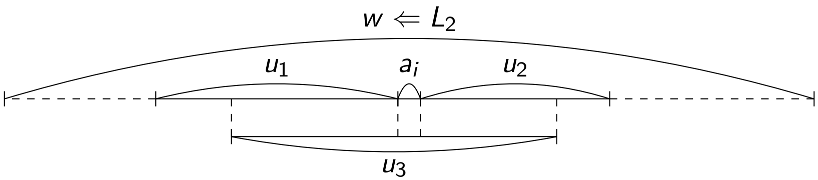
1. *if $k = 2$, then $aa \notin F(w)$;*
2. *otherwise, $w \in (\Sigma \setminus \{a\})^*$.*

Define $M_u = \{a \in \Sigma \mid u \in \mathrm{Suff}(u) a \mathrm{Pref}(u)\}$. By definition, $M_u \neq \emptyset$ if and only if $u \notin X$.

Proposition

Let $u \in Q_B^{(=1)}$. Then $(w \Leftarrow u) \Rightarrow u = \{w\}$ for $w \in \Sigma^+$ if and only if $w \in (\Sigma \setminus M_u)^+$.

$(L_1 \Leftarrow L_2) \Rightarrow L_2 = L_1$?



where $w \in L_1$, and u_1, u_2, $u_3 \in L_2$.

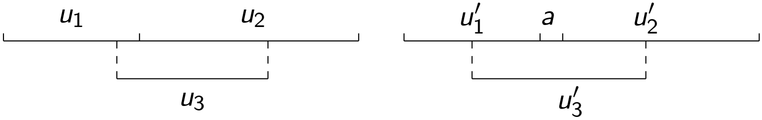
Definition

A set $L \subseteq \Sigma^+$ is called a <u>comma code</u> if $L\Sigma L \cap \Sigma^+ L \Sigma^+ = \emptyset$.

Theorem

If language $L_2 \subseteq \Sigma^+$ is a comma code, the equation $(L_1 \Leftarrow L_2) \Rightarrow L_2 = L_1$ holds for any language $L_1 \subseteq \Sigma^*$.

Comma-free Codes vs. Comma Codes



$L^2 \cap \Sigma^+ L \Sigma^+ = \emptyset$ $\qquad$ $L \Sigma L \cap \Sigma^+ L \Sigma^+ = \emptyset$

Lemma

For a language $L \subseteq \Sigma^*$, L is a comma code if and only if $L\Sigma$ is a comma-free code.

Recall: $L \subseteq \Sigma^+$ is an infix code if $L \cap (\Sigma^* L \Sigma^+ \cup \Sigma^+ L \Sigma^*) = \emptyset$.

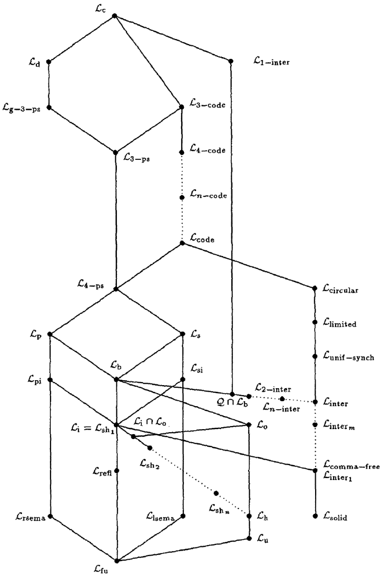
Lemma

For a language $L \subseteq \Sigma^*$, L is an infix code if and only if $L\Sigma$ is an infix code.

Corollary

A comma code is an infix code, and hence a code.

Relations Between Families of Languages or Codes [4]



- $\mathcal{L}_i$: infix codes
- $\mathcal{L}_{comma} \subseteq \mathcal{L}_i$
- $\mathcal{L}_{comma\text{-}free} \subset \mathcal{L}_i$

Comma-free Codes vs. Comma Codes

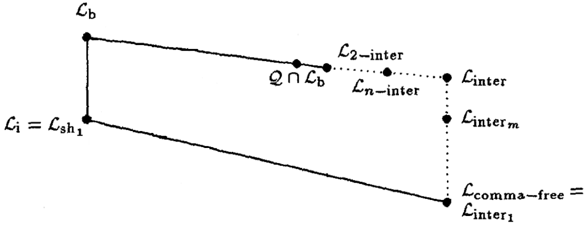
Proposition

The family of comma codes and the family of comma-free codes are incomparable, but not disjoint.

Example

1) Consider $L_1 = \{aba, abba\}$. It is a comma-free code, but $abababa \in L\Sigma L \cap \Sigma^+ L \Sigma^+$.
2) Consider $L_2 = \{aaab, abab\}$. It is a comma code, but $abababab \in L^2 \cap \Sigma^+ L \Sigma^+$.
3) Consider $L_3 = \{abba, abbba\}$.

Relations Between Families of Languages or Codes [4]

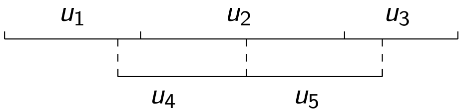


- $\mathcal{L}_{comma\text{-}free} \subset \mathcal{L}_i$
- $\mathcal{L}_{inter_1} \subset \cdots \subset \mathcal{L}_{inter_m} \subset \cdots \subset \mathcal{L}_b$ (Bifix codes)

Definition

For $m \geq 1$, a nonempty set $L \subseteq \Sigma^+$ is called an intercode of index m if $L^{m+1} \cap \Sigma^+ L^m \Sigma^+ = \emptyset$.

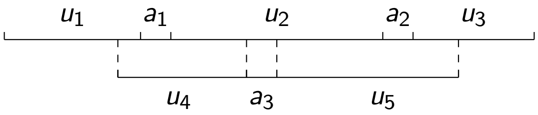
e.g. if $m = 2$, $u_1, u_2, u_3, u_4, u_5 \in L$



Definition

For $m \geq 1$, a nonempty set $L \subseteq \Sigma^+$ is called a <u>comma intercode of index m</u> if $(L\Sigma)^m L \cap \Sigma^+ (L\Sigma)^{m-1} L \Sigma^+ = \emptyset$.

e.g. $m = 2$, $u_1, u_2, u_3, u_4, u_5 \in L$



Properties of Comma Intercodes

Proposition

A comma intercode is a bifix code, and hence a code.

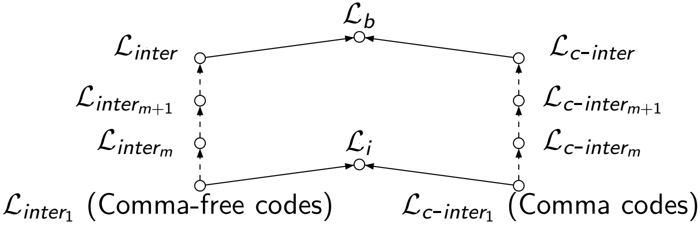
Theorem

$\mathcal{L}_{c\text{-}inter_1} \subset \mathcal{L}_{c\text{-}inter_2} \subset \cdots \subset \mathcal{L}_{c\text{-}inter_m} \subset \cdots \subset \mathcal{L}_b$ *holds.*

Example

$L = \{aba, abba\}$ is a bifix code, but not a comma intercode.

Relationship Between Intercodes and Comma Intercodes



Recall

The family of comma codes and the family of comma-free codes are incomparable.

Corollary

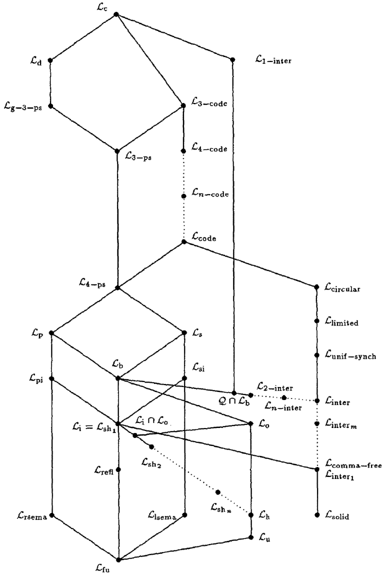
For any $m, n \geq 1$*, the family of intercodes of index* m *and the family of comma intercode of index* n *are incomparable.*

Conclusion

- Complete condition for $(w \Leftarrow u) \Rightarrow u = \{w\}$
- Conditions for $(L_1 \Leftarrow L_2) \Rightarrow L_2 = L_1$
- Comma codes and their properties
- Comma intercodes and their properties
- An new infinite hierarchy between infix codes and bifix codes

Recent Results and Future Works

- Generalization of Comma codes
  $L\Sigma^k L \cap \Sigma^+ L \Sigma^+ = \emptyset$
- Generalization of Comma codes
  $L(\Sigma \cup \cdots \cup \Sigma^k) L \cap \Sigma^+ L \Sigma^+ = \emptyset$



References

- Berstel, J., Perrin, D.: Theory of Codes. Academic Press. Inc., Orlando, Toronto. (1985)
- Domaratzki, M.: Deletion along trajectories. Theoretical Computer Science. **320** (2004) 293-313
- Ito, M., Kari, L., Thierrin, G.: Insertion and deletion closure of languages. Theoretical Computer Science. **183** (1997) 3-19
- Jürgensen, H., Konstantinidis, S.: The hierarchy of codes. Proc. of FCT 1993, LNCS. **710** (1993) 50-68
- Kari, L.: On Insertion and Deletion in Formal Languages. Ph.D. Thesis, University of Turku. (1991)
- Kari, L.: Insertion and deletion of words: determinism and reversibility. Proc. of MFCS 1992, LNCS. **629** (1992) 315-326
- Kari, L., Thierrin, G.: Words insertions and primitivity. Utilitas Mathematica. **53** (1998) 49-61
- Kari, L., Mateescu, A., Paun, Gh., Salomaa, A.: Deletion sets. Fundamenta Informatica. **18**, 1 (1993) 355-370
- Kari, L., Mateescu, A., Paun, Gh., Salomaa, A.: On parallel deletions applied to a word. RAIRO, Theoret. Inform. Appl. **29** (1995) 129-144
- Kudlek, M., Mateescu, A.: On distributed catenation. Theoretical Computer Science. **180** (1997) 341-352
- Kudlek, M., Mateescu, A.: On mix operation. New Trends in Formal Languages, G. Păun and A. Salomaa, Eds., LNCS. **1218** (1997) 35-44
- Lothaire, M.: Algebraic Combinatorics on Words, Cambridge University Press, 2002
- Manea, F., Mitrana, V., Yokomori, T.: Two complementary operations inspired by the DNA hairpin formation: Completion and reduction. Theoretical Computer Science. **410** (2009) 417-425
- Mateescu, A., Rozenberg, G., Salomaa, A.: Shuffle on trajectories: syntactic constraints. Theoretical Computer Science. **197** (1998) 1-56
- Parikh, R.J.: On context-free languages. Journal of the Association for Computing Machinery. **13** (1966) 570- 581
- Shyr, H. J.: Free Monoids and Languages. Lecture Notes, Institute of Applied Mathematics, National Chung-Hsing University, Taichung, Taiwan. (2001)
- Yu, S. S.: Languages and Codes. Lecture Notes, Department of Computer Science, National Chung-Hsing University, Taichung, Taiwan 402. (2005)
- Yu, S. S.: A characterization of intercodes. Intern. J. Computer Math. **36** (1990) 39-48