Conway Factors and the Carrez NFA

K. Sutner

2024/01/07 v.0.2

1 Conway Factorizations

Fix some alphabet Σ once and for all; all languages in the following will be subsets of $\Sigma^\star$. We are mostly interested in regular languages, but if a notion makes sense in a the setting of all languages we will define it in full generality. The following definition is due to Conway [3].

Definition 1.1 *A* k-subfactorization *of* L, $k \geq 2$, *(or subfactorization of order* k*) is a* k*-tuple of languages* X_i, $1 \leq i \leq k$, *such that*

$$X_1 \cdot X_2 \cdot \ldots X_{k-1} \cdot X_k \subseteq L$$

A k-factorization *is a* k*-subfactorization where every term is maximal.*

For emphasis, we write $\boldsymbol{X} = X_1{:}X_2{:}\ldots{:}X_k$ for a subfactorization and refer to X_i as the ith *term* or ith *component* of $\boldsymbol{X}$. As usual, we will mostly express concatenation by juxtaposition and write XY rather than $X{\cdot}Y$. Note that $\ldots{:}X{:}Y{:}\ldots$ is a subfactorization iff its *contraction* $\ldots{:}XY{:}\ldots$ is a subfactorization, albeit of order $k-1$. Alas, the corresponding claim for factorizations is wrong, in either direction.

There is a natural partial order on k-subfactorizations by pointwise set inclusion: $Y_1{:}Y_2{:}\ldots{:}Y_k \sqsubseteq X_1{:}X_2{:}\ldots{:}X_k$ if $Y_i \subseteq X_i$ for all i. So a factorization is a maximal element in this order. We write $\mathcal{F}_k$ for the poset of all k-subfactorizations and $\widehat{\mathcal{F}}_k$ for all k-factorizations.

Definition 1.2 *A* factor *of* L *is a term that appears in some place in some factorization. A* left/right factor *is one that appears in the first/last position of a factorization.*

Contractions show that all left/right factors already appear in 2-factorizations, so it is natural to study $\widehat{\mathcal{F}}_2$ first. For any subset X of $\Sigma^\star$, consider X to be a term in a 2-subfactorization and saturate the other term to obtain a factor. More precisely, the corresponding left and right factors are

$$\lambda_L X = \{\, u \in \Sigma^\star \mid uX \subseteq L \,\} = \bigcup\{\, Z \mid ZX \subseteq L \,\}$$
$$\rho_L X = \{\, u \in \Sigma^\star \mid Xu \subseteq L \,\} = \bigcup\{\, Z \mid XZ \subseteq L \,\}$$

and we have subfactorizations $\lambda_L X{:}X, X{:}\rho_L X \subseteq L$. The maps λ_L and ρ_L are anti-monotonic by definition and can naturally be expressed in terms of left and right quotients:

$$\lambda X = \{\, u \in \Sigma^\star \mid X \subseteq u^{-1}L \,\}$$
$$\rho X = \{\, u \in \Sigma^\star \mid X \subseteq Lu^{-1} \,\}$$

The key property that we will use without further mention is

$$XY \subseteq L \iff X \subseteq \lambda Y, Y \subseteq \rho X$$

To lighten notation, we will omit the subscripts in ρ_L and λ_L whenever the language is clear from context.

For some simple examples, first let $L = \Sigma^\star$. Then all factorizations have only terms $\Sigma^\star$. Second, for $L = \emptyset$ every factorization must contain exactly one term 0, all others are $\Sigma^\star$. Thus, there are exactly two 2-factorizations: $0{:}\Sigma^\star$ and $\Sigma^\star{:}0$. We have $\lambda\Sigma^\star = \rho\Sigma^\star = 0$ and $\lambda 0 = \rho 0 = \Sigma^\star$. Lastly, for $L = a^+ \subseteq \{a, b\}^\star$ we have $\lambda(0) = \Sigma^\star$, $\lambda(\varepsilon) = a^+$, $\lambda(a^+) = a^\star$ and $\lambda(\Sigma^\star) = 0$, and these are all the left and right factors.

Writing op for the reversal of strings, the two operations are connected by $\lambda_L X = \left(\rho_{L^{\mathsf{op}}} X^{\mathsf{op}}\right)^{\mathsf{op}}$ and similarly the other way around. The following lemma shows that the language L itself appears as a left and as a right factor.

Lemma 1.1 *For all $X \subseteq \Sigma^\star$: $X \subseteq \rho\lambda X$ and $X \subseteq \lambda\rho X$. For $X = L$ we have equality.*

Proof. By definition, $\lambda X \cdot X \subseteq L$, so $X \subseteq \rho\lambda X$. When $X = L$ and $u \in \rho\lambda L$, then $u \in L$ since $\varepsilon \in \lambda L$. The second part follows immediately from the first and the observation preceding the lemma. □

Lemma 1.2 *There is a one-one correspondence between all left factors and all right factors. In fact, the correspondence is given by the maps λ and ρ.*

Proof. Suppose X is a left factor and let $X{:}Y$ be a corresponding factorization. Then $Y \subseteq \rho X$ and, since Y is maximal, we must have $Y = \rho X$. Similarly $X = \lambda Y$ and we are done. □

Turning to arbitrary factors, suppose that Z is a factor that appears in the ith position of some k-factorization, $1 < i < k$. By merging the other components we obtain a 3-subfactorization $X'{:}Z{:}Y'$ which can be extended to a 3-factorization $X{:}Z{:}Y$. Thus is suffices to characterize the middle terms of 3-factorizations. To this end, let X be a left factor and Y a right factor. Since $X{:}0{:}Y$ is a subfactorization, there is a unique set Z such that $X{:}0{:}Y \sqsubseteq X{:}Z{:}Y$, which set we will denote $Z(X, Y)$.

Lemma 1.3 *Let $X' = \lambda(L)$ and $Y' = \rho(L)$. Then all left factors are of the form $Z(X', Y)$ where Y is a right factor, and all right factors are of the form $Z(X, Y')$ where X is a left factor. Furthermore, $Z(X', Y') = L$.*

Proof. By our choice of X', $X'{:}\lambda(Y){:}Y$ is a factorization for all right factors Y. There argument for right factors is analogous. Since $X'{:}L{:}Y'$ is a factorization, we have $Z(X', Y') = L$. □

Theorem 1.1 *The number of factors of L is finite if, and only if, L is regular. Moreover, the number of left/right factors is $\widehat{\mu}(\overline{L})$ in this case.*

Proof. By lemma 1.3 it suffices to consider only left/right factors. Accordingly, let $X{:}Y$ be any factorization of L. Then

$$Y = \rho X = \bigcap_{u \in X} u^{-1}L = \overline{X^{-1}\overline{L}}$$

But L is regular iff $\overline{L}$ is regular iff the number of quotients, word or language, is finite. We are done by lemma 1.1. □

Note, though, that $\widehat{\mu}(L) \neq \widehat{\mu}(\overline{L})$ in general, in distinction to words quotients. It follows that for regular L there are at most $2^{\mu(L)}$ many left/right pairs. We will consider a direct construction of a finite state machine for L based on left factors in the next section.

Not let L be regular, say, there are m left/right factors and we have chosen some linear ordering of these languages. This affords a coordinate system and we can organize the collection of all factors into a $m \times m$ matrix $\mathfrak{F}$ with entries $Z_{ij} = Z(X_i, Y_j)$.

Example 1.1 The star-free language $L = a^\star b^\star c^\star$ has 5 left/right factors:

left	$\Sigma^\star$	L	$a^\star b^\star$	$a^\star$	0
right	0	$c^\star$	$b^\star c^\star$	L	$\Sigma^\star$

and the corresponding factor matrix $\mathfrak{F} = (Z_{ij})$ looks like so:

	0	$c^\star$	$b^\star c^\star$	L	$\Sigma^\star$
$\Sigma^\star$	$\Sigma^\star$	0	0	0	0
L	$\Sigma^\star$	$c^\star$	0	0	0
$a^\star b^\star$	$\Sigma^\star$	$b^\star c^\star$	$b^\star$	0	0
$a^\star$	$\Sigma^\star$	L	$a^\star b^\star$	$a^\star$	0
0	$\Sigma^\star$	$\Sigma^\star$	$\Sigma^\star$	$\Sigma^\star$	$\Sigma^\star$

Theorem 1.2 *Consider the $m \times m$ factor matrix $\mathfrak{F} = (Z_{ij})$ of some regular language L. Then*

1. $Z_{ij}Z_{jk} \subseteq Z_{ik}$
2. *X_1:X_2:$\ldots$:X_s is a subfactorization iff there is an index sequence $1 \leq i_0, \ldots, i_s \leq m$ such that $i_0 = \lambda(L)$, $X_j \subseteq Z_{i_{j\text{-}1}\, i_j}$ and $i_s = \rho(L)$.*

Proof. By definition, $X_i Z_{ij} Y_j \subseteq L$, so that $X_i Z_{ij} \subseteq X_j$. Hence $X_i Z_{ij} Z_{jk} Y_k \subseteq X_j Z_{jk} Y_k \subseteq L$, and our claim follows.

For the second part, it suffices to prove the binary case: $XY \subseteq Z_{ik}$ iff there is some j such that $X \subseteq Z_{ij}$ and $Y \subseteq Z_{jk}$. To see this, note that $(X_i X)(Y Y_k) \subseteq L$, so that $X_i X \subseteq X_j$ and $Y Y_j \subseteq Y_j$ for some j. But then $X_i X Y_j \subseteq L$ and $X_J Y Y_k \subseteq L$, and the claim follows. □

Back to our original complaint: the lack of invariance of state complexity under string reversal.

Theorem 1.3 (Conway) *Let L be a regular language. Then $\widehat{\mu}(L) = \widehat{\mu}(L^{\mathsf{op}})$.*

Proof. Consider all 2-factorizations X:Y of $\overline{L}$. As we have just seen, there are $\widehat{\mu}(L)$ choices for X. By symmetry, there are $\widehat{\mu}(L^{\mathsf{op}})$ choices for Y. But we already know that these two numbers agree. □

1.1 Computation

Let L be a regular language and $\mathcal{M} = \langle Q, \Sigma, \delta; q_0, F\rangle$ it's minimal DFA, so the state complexity of $\mathcal{M}$ is $\mu(L)$. We write δ as a right semigroup action $p \cdot x = \delta(p, x)$. All behaviors and cobehaviors below are with respect to $\mathcal{M}$. Suppose we have m left/right factor pairs X:Y. From the proof of theorem 1.1 we have

$$Y = \rho X = \bigcap_{u \in X} u^{-1} L$$

Similarly we can express λ as follows. The quotient map $x \mapsto x^{-1}L$ induces an equivalence relation on $\Sigma^\star$; write $[u]$ for the equivalence class of u. Hence $[u] = [\![\delta(q_0, u)]\!]^{\mathsf{co}}$ and

$$X = \lambda Y = \bigcup_{Y \subseteq u^{-1}L} [u]$$

In terms of the minimal DFA this means

$$Y = \bigcap \{\, [\![q_0 \cdot u]\!] \mid u \in X \,\}$$
$$X = \bigcup \{\, [\![p]\!]^{\mathsf{co}} \mid Y \subseteq [\![p]\!] \,\}$$

Letting $P = \{\, q_0 \cdot u \mid u \in X \,\} \subseteq Q$ we have $Y = \bigcap_{p \in P} [\![p]\!]$. We call P *critical* if P produces Y in this manner, and P is maximal such (and thus actually maximum). Given P critical we have $X = [\![P]\!]^{\mathsf{co}}$. Hence we can construct a list

$$X_1{:}Y_1, X_2{:}Y_2, \ldots, X_m{:}Y_m$$

of all left/right pairs by computing the critical state sets $P \subseteq Q$.

Suppose $P_1, P_2 \subseteq Q$ are critical, and let X be the left factor for P_1, and Y the right factor for P_2. To determine the middle factor $Z = Z(X, Y)$, note that

$$\begin{aligned} u \notin Z &\Leftrightarrow \exists\, x \in X, y \in Y\, (xuy \notin L) \\ &\Leftrightarrow \exists\, q \in P_1, y \in Y\, (q \cdot uy \notin F) \\ &\Leftrightarrow \exists\, q \in P_1\, (Y \not\subseteq [\![q \cdot u]\!]) \\ &\Leftrightarrow \exists\, q \in P_1\, (q \cdot u \notin P_2) \end{aligned}$$

But then $\overline{Z}$ is the language of $\mathcal{M}(P_1, \overline{P_2})$ and $Z = \bigcap_{p \in P_1} [\![\mathcal{M}(p, \overline{P_2})]\!]$.

2 The Carrez Automaton

For any NFA $\mathcal{A}$ and state p, write $[\![p]\!]_{\mathcal{A}}$ for the behavior of p (i.e., the language of $\mathcal{A}(p, F)$) and write $[\![p]\!]^{\mathsf{co}}_{\mathcal{A}}$ for the co-behavior of p (i.e., the language of $\mathcal{A}(I, p)$). We omit subscripts when the automaton in question is obvious. A *homomorphism* of NFAs is any map that preserves transitions, initial and final states. Hence, for any homomorphism $h : \mathcal{A} \to \mathcal{B}$, $[\![p]\!]_{\mathcal{A}} \subseteq [\![h(p)]\!]_{\mathcal{B}}$ and $[\![p]\!]^{\mathsf{co}}_{\mathcal{A}} \subseteq [\![h(p)]\!]^{\mathsf{co}}_{\mathcal{B}}$. As a consequence, $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$. We may safely assume that all NFA are trim.

The following definition is due to Christian Carrez [2] and dates back to 1970; the description here is based on [1].

Definition 2.1 *For any language L, define the* Carrez automaton *for L, in symbols $\mathcal{C}_L$, as follows:*

1. *states:* $Q = \{\, \lambda X \mid X \subseteq \Sigma^\star, X, \lambda X \neq \emptyset \,\}$
2. *initial:* $I = \{\, Z \in Q \mid \varepsilon \in Z \,\}$
3. *final:* $F = \{\, Z \in Q \mid Z \subseteq L \,\}$
4. *transitions:* $Z \xrightarrow{a} Z' \iff Z \cdot a \subseteq Z'$

By induction, $X \xrightarrow{u} Y \Leftrightarrow X \cdot u \subseteq Y$, so that $\emptyset \xrightarrow{u} Z \xrightarrow{u} \Sigma^\star$.

Lemma 2.1 *The Carrez automaton $\mathcal{C}_L$ accepts the language L.*

Proof.

From the definitions, λL is initial and $\lambda\rho L = L$ is final. But $(\lambda L) \cdot L \subseteq L$, so $\lambda L \xrightarrow{u} L$ is an accepting computation in $\mathcal{C}_L$ for all $u \in L$. On the other hand, whenever $\mathcal{C}_L$ accepts u, we have a computation $X \xrightarrow{u} Y$ where $\varepsilon \in X$ and $Y \subseteq L$. But $u \in Y$ and we are done. □

Another way to show that $L \subseteq \mathcal{L}(\mathcal{C}_L)$ is to note that there are transitions of the form $\lambda X \xrightarrow{a} \lambda\, a^{-1}X$. Hence an accepting computation in the quotient automaton for L translates into a computation $\lambda L \xrightarrow{u} \lambda u^{-1}L$ in $\mathcal{C}_L$. But λL is initial, and $\varepsilon \in u^{-1}L$, whence $\lambda u^{-1}L \subseteq L$ is final, and we have an accepting computation in $\mathcal{C}_L$. More generally, the states of $\mathcal{C}_L$ are closely related to their own behaviors and cobehaviors.

Lemma 2.2 *Consider any state Z in the Carrez automaton $\mathcal{C}_L$. Then $[\![Z]\!]^{\mathsf{co}} = Z = \lambda[\![Z]\!]$. Similarly $[\![Z]\!] = \rho[\![Z]\!]^{\mathsf{co}}$.*

Proof. $u \in [\![Z]\!]^{\mathsf{co}}$ implies that there is a computation from some initial state $\varepsilon \in \lambda Y$ to Z, hence $u \in Z$. Conversely, if $u \in Z = \lambda X$, consider the initial state λL. Since $\lambda L \cdot L \subseteq L$ and $uX \subseteq L$, we have $(\lambda L) \cdot (uX) \subseteq L$. But then $(\lambda L) \cdot u \in \lambda X$, hence $u \in [\![Z]\!]^{\mathsf{co}}$.

By the definition of $\mathcal{C}_L$, $Z \cdot [\![Z]\!] \subseteq L$, so $Z \subseteq \lambda[\![Z]\!]$. But $\lambda X \cdot X \subseteq L$, so $X \subseteq [\![\lambda X]\!] = [\![Z]\!]$, whence $\lambda[\![Z]\!] \subseteq \lambda X = Z$.

The second claim is entirely similar. □

As a consequence of the last lemma, $\mathcal{C}_L$ is rigid in the sense that it admits no non-trivial endomorphisms.

Corollary 2.1 *The only endomorphism of $\mathcal{C}_L$ is the identity.*

Proof. Let h be an endomorphism of $\mathcal{C}_L$ and $h(X) = Y$, so that $[\![X]\!] \subseteq [\![Y]\!]$. By anti-monotonicity, $Y = \lambda[\![Y]\!] \subseteq \lambda[\![X]\!] = X$. A similar argument with co-behaviors shows $X \subseteq Y$. □

The following result is now a direct consequence of theorem 1.1.

Theorem 2.1 (Carrez) *L is regular iff $\mathcal{C}_L$ has a finite number of states.*
Moreover, if n is the state complexity of L, then the state complexity of $\mathcal{C}_L$ is at most 2^n.

Example 2.1 If $L = a^+$ and $\Sigma = \{a\}$, then the number of states in $\mathcal{C}_L$ is the same as the number of states in the minimal DFA for L, namely 2. If $\Sigma = \{a, b\}$ and L is the set of words with an even number of both a's and b's, then $\mathcal{C}_L$ has 5 states.

Now let $\mathcal{A} = \langle Q, \Sigma, \tau; I, F\rangle$ be any trim NFA which accepts a sublanguage of L, and define a map h from $\mathcal{A}$ to $\mathcal{C}_L$ by:

$$h(p) = \lambda[\![p]\!]_{\mathcal{A}}$$

We assume that $\mathcal{A}$ is trim to avoid rogue states.

Lemma 2.3 *The function h is a homomorphism from $\mathcal{A}$ to $\mathcal{C}_L$.*

Proof. Let $p \xrightarrow{a} q$ be a transition in $\mathcal{A}$ and $u \in h(p) = \lambda[\![p]\!]$. Then $u[\![p]\!] \subseteq L$. Since $a[\![q]\!] \subseteq [\![p]\!]$, we have $ua \in \lambda[\![q]\!]$. Thus $\mathcal{C}_L$ has a transition $\lambda[\![p]\!] \xrightarrow{a} \lambda[\![q]\!]$. If p is initial in $\mathcal{A}$, then $[\![p]\!]_{\mathcal{A}} \subseteq L$ and therefore $\varepsilon \in \lambda[\![p]\!]_{\mathcal{A}}$; hence $h(s)$ is initial in $\mathcal{C}_L$. Lastly, suppose p is final in $\mathcal{A}$, whence $\varepsilon \in [\![p]\!]_{\mathcal{A}}$. But then $u \in \lambda[\![p]\!]_{\mathcal{A}}$ implies $u \in L$ and $h(p) = \lambda[\![p]\!]_{\mathcal{A}}$ is also final in $\mathcal{C}_L$. □

Combining the last two results produces the theorem in [1]: no proper homomorphic image of $\mathcal{C}_L$ can accept L, or even a subset thereof.

Theorem 2.2 *Let $\mathcal{A}$ be and NFA accepting a subset of L and $g : \mathcal{C}_L \to \mathcal{A}$ an epimorphism. Then g is already an isomorphism.*

Proof. We have seen that the behavioral map provides a homomorphism $h : \mathcal{A} \to \mathcal{C}_L$. Hence $g \circ h$ is an endomorphism of $\mathcal{C}_L$, and thus the identity. Hence g is an isomorphism. □

References

[1] André Arnold, Anne Dicky, and Maurice Nivat. A note about minimal non-deterministic automata. *Bulletin of the EATCS*, 47:166–169, 01 1992.

[2] Christian Carrez. On the minimalization of non-deterministic automaton, 1970.

[3] John Conway. *Regular Algebra and Finite Machines*. Chapman and Hall, 1971.