Graduate Program in Computer Science, Universidade Federal de Santa Catarina
INE410113 (Theory of Computation)

Converting NFAs to regular expressions with the Brzozowski algebraic method

Gustavo Zambonin

An arguably simple method to convert non-deterministic finite automata (NFA) to regular expressions is due to Brzozowski [Brz64]. It expresses the automaton as a system of equations, in which each equation describes the language accepted by a particular state of the automaton. The objective is to solve the system until only a single equation is left, featuring a closed form regular expression.

Consider a NFA $A = (Q, \Sigma, \delta, q_0, F)$ free of ε-transitions, and in this context, operations of union ($\cup$), concatenation ($\cdot$) and Kleene star (*). Associativity and distributivity of union and concatenation, and commutativity of union are desired properties for this method to work. Furthermore, we will observe shortly that the systems generated may only be solved if we also take into account the lemma below.

Arden's lemma [Ard61]. Let $L, U, V \subseteq \Sigma^*$ be regular languages with $\varepsilon \notin U$. Then,

$$L = UL \cup V \Leftrightarrow L = U^*V. \tag{1}$$

The description for the method is as follows. For every state $q_i \in Q$, create the equation

$$Q_i = \bigcup_{q_i \xrightarrow{a} q_j} aQ_j \cup \begin{cases} \{\varepsilon\}, & q_i \in F \\ \emptyset, & q_i \notin F, \end{cases} \tag{2}$$

and solve the system for Q_0, using the properties of operations listed above, as well as Eq. 1.

Example. Consider the regular language $\mathcal{L} = \{w \mid w \in \Sigma^* \wedge \texttt{110110} \text{ is not a substring of } w\}$.

We will first need to build an automaton which accepts this language. Note that we can build $\overline{\mathcal{L}}$, that is, the language that accepts every string which contains 110110, and then invert accepting and non-accepting states. This is allowed, since regular languages are closed under complement. We present the output of this rationale below.

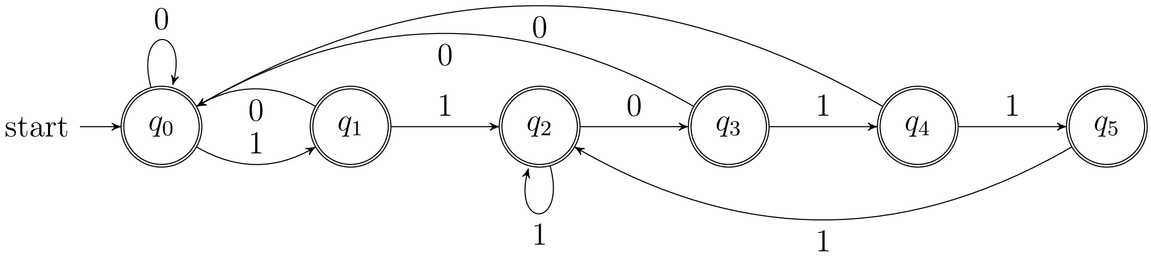

Figure 1: NFA that accepts $\mathcal{L}$.

Now, we apply the method above to convert the automaton into a system of equations. By Eq. 2, we get

$$Q_0 = \varepsilon + 0Q_0 + 1Q_1, \tag{3}$$
$$Q_1 = \varepsilon + 0Q_0 + 1Q_2, \tag{4}$$
$$Q_2 = \varepsilon + 0Q_3 + 1Q_2, \tag{5}$$
$$Q_3 = \varepsilon + 0Q_0 + 1Q_4, \tag{6}$$
$$Q_4 = \varepsilon + 0Q_0 + 1Q_5, \tag{7}$$
$$Q_5 = \varepsilon + 1Q_2. \tag{8}$$

By substituting Eq. 8 into Eq. 7, we get

$$Q_4 = \varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2). \tag{9}$$

By substituting Eq. 9 into Eq. 6, we get

$$Q_3 = \varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2)). \tag{10}$$

By substituting Eq. 10 into Eq. 5, we get

$$Q_2 = \varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2))) + 1Q_2. \tag{11}$$

Finally, by substituting Eq. 4 into Eq. 3, we get

$$Q_0 = \varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1Q_2). \tag{12}$$

Our system is resumed to Eq. 12 and Eq. 11, described in terms of themselves and each other. We then apply Eq. 1 to Eq. 12, with $L = Q_0,\ U = 0,\ V = \varepsilon + 1(\varepsilon + 0Q_0 + 1Q_2)$, and we get

$$Q_0 = 0^*(\varepsilon + 1(\varepsilon + 0Q_0 + 1Q_2)). \tag{13}$$

We use Eq. 1 on Eq. 11, with $L = Q_2,\ U = 1,\ V = \varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2)))$, and we get

$$Q_2 = 1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2)))). \tag{14}$$

Both equations will not allow further use of Eq. 1 without manual intervention. Hence, we expand Eq. 14 to permit this.

$$\begin{align} Q_2 &= 1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1(\varepsilon + 1Q_2)))) \tag{15}\\ &= 1^*0111Q_2 + 1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1))). \tag{16} \end{align}$$

We use Eq. 1 on Eq. 16 with $L = Q_2,\ U = 1^*0111,\ V = 1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1)))$ to remove its last circular definition, obtaining

$$Q_2 = (1^*0111)^*(1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1)))). \tag{17}$$

Now, we are able to substitute Eq. 17 into Eq. 13, and expand it, so Eq. 1 may be used.

$$\begin{align} Q_0 &= 0^*(\varepsilon + 1(\varepsilon + 0Q_0 + 1(1^*0111)^*(1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1)))))) \tag{18}\\ &= 0^*10Q_0 + 0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*(1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1)))))) \tag{19} \end{align}$$

We use Eq. 1 on Eq. 19 with $L = Q_0,\ U = 0^*10$,
$V = 0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*(1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1))))))$, and expand it again, obtaining

$$\begin{align} Q_0 &= (0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 0Q_0 + 1(\varepsilon + 0Q_0 + 1))))) \tag{20}\\ &= (0^*10)^*0^*11(1^*0111)^*1^*00Q_0 \nonumber\\ &\quad + (0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 0Q_0 + 1))))). \tag{21} \end{align}$$

We use Eq. 1 on Eq. 21 with $L = Q_0$, $U = (0^*10)^*0^*11(1^*0111)^*1^*00$, $V = (0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 0Q_0 + 1)))))$, and expand it for the last time, obtaining

$$\begin{aligned} Q_0 &= ((0^*10)^*0^*11(1^*0111)^*1^*00)^* \\ &\quad \cdot (0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 0Q_0 + 1))))) \end{aligned} \tag{22}$$

$$\begin{aligned} &= ((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*11(1^*0111)^*1^*010Q_0 \\ &\quad + ((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 1))))). \end{aligned} \tag{23}$$

Finally, we use Eq. 1 on Eq. 23 with $L = Q_0$,
$U = ((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*11(1^*0111)^*1^*010$,
$V = ((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 1)))))$, and obtain a closed form regular expression:

$$\begin{aligned} Q_0 &= (((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*11(1^*0111)^*1^*010)^* \\ &\quad \cdot ((0^*10)^*0^*11(1^*0111)^*1^*00)^*(0^*10)^*0^*(\varepsilon + 1(\varepsilon + 1(1^*0111)^*1^*(\varepsilon + 0(\varepsilon + 1(\varepsilon + 1))))). \end{aligned} \tag{24}$$

References

[Ard61] D. N. Arden. Delayed-logic and finite-state machines. In R. S. Ledley, editor, *2nd Annual Symposium on Switching Circuit Theory and Logical Design*, pages 133–151, October 1961.

[Brz64] J. A. Brzozowski. Derivatives of regular expressions. *Journal of the ACM*, 11(4):481–494, October 1964.