Chapter II ω-Automata

1. ω-Automata and ω-Languages

- ω-automata accept (or reject) words of infinite length
- ω-languages consisting of infinite words appear:
  - in verification, as encodings of non-terminating executions of a program
  - in arithmetic, as encodings of sets of real numbers

ω-Languages

- An ω-word is an infinite sequence of letters.
- The set of all ω-words is denoted by Σ^ω.
- An ω-language is a set of ω-words, i.e., a subset of Σ^ω.
- A language L_1 can be concatenated with an ω-language L_2 to yield the ω-language $L_1 L_2$, but two ω-languages cannot be concatenated.
- The ω-iteration of a language $L \subseteq \Sigma^\star$, denoted by L^ω, is an ω-language.
- Observe: $\emptyset^\omega = \emptyset$.

ω-Regular Expressions

- ω-regular expressions have syntax

$$s ::= r^{\omega} \mid r s_1 \mid s_1 + s_2$$

  where r is an (ordinary) regular expression.

- The ω-language $L_{\omega}(s)$ of an ω-regular expression s is inductively defined by

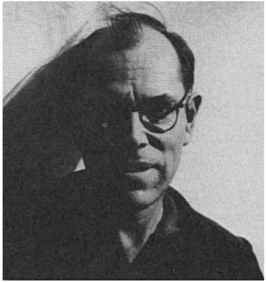
$$L_{\omega}(r^{\omega}) = \left(L(r)\right)^{\omega} \quad L_{\omega}(r s_1) = L(r) L_{\omega}(s_1)$$

$$L_{\omega}(s_1 + s_2) = L_{\omega}(s_1) \cup L_{\omega}(s_2)$$

- A language is ω-regular if it is the language of some ω-regular expression .

Büchi Automata

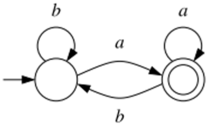
- Invented by J.R. Büchi, swiss logician.

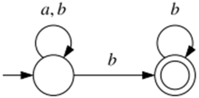


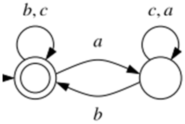
Büchi Automata

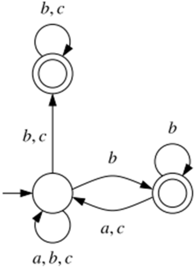
- Same syntax as DFAs and NFAs, but different acceptance condition.
- A run of a Büchi automaton on an ω-word is an infinite sequence of states and transitions.
- A run is accepting if it visits the set of final states infinitely often.
  - Final states renamed to accepting states.
- A DBA or NBA A accepts an ω-word if it has an accepting run on it; the ω-language $L_\omega(A)$ of A is the set of ω-words it accepts.

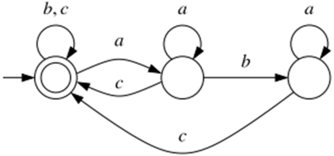
Some examples



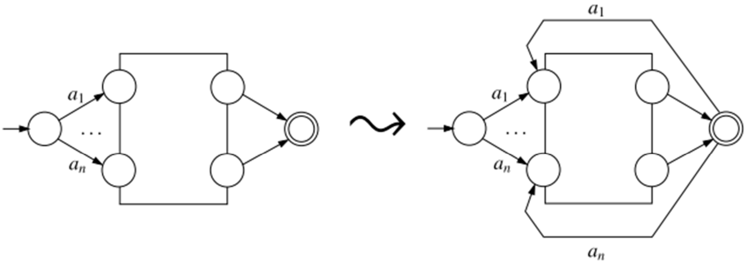






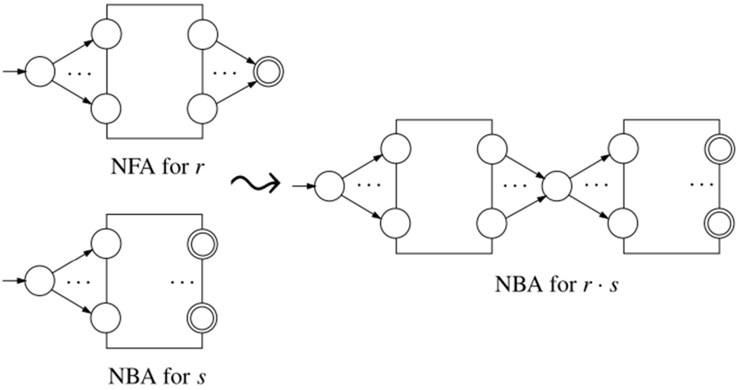


From ω-Regular Expressions to NBAs

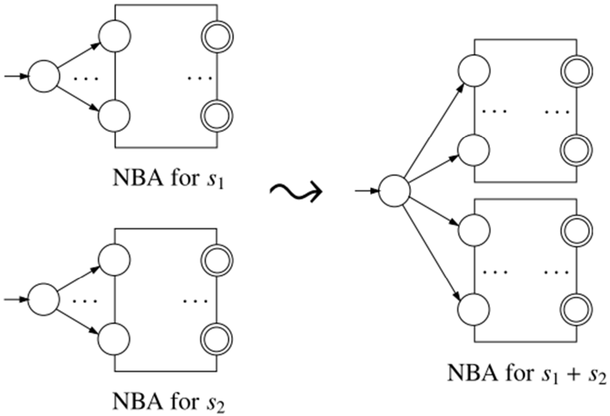


NFA for r NBA for r^ω

From ω-Regular Expressions to NBAs



From ω-Regular Expressions to NBAs

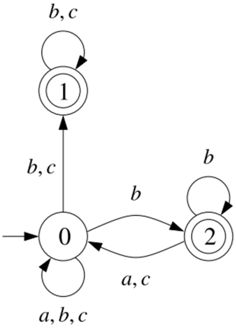


From NBAs to ω-Regular Expressions

- Lemma: Let A be a NFA, and let q, q' be states of A . The language $L_q^{q'}$ of words with runs leading from q to q' and visiting q' exactly once is regular.
- Let $r_q^{q'}$ denote a regular expression for $L_q^{q'}$.

From NBAs to ω-Regular Expressions

- Example:



$$
\begin{aligned}
r_0^1 &= (a+b+c)^*(b+c) \\
r_0^2 &= (a+b+c)^*b \\
r_1^1 &= (b+c)^* \\
r_2^2 &= b+(a+c)(a+b+c)^*b
\end{aligned}
$$

From NBAs to ω-Regular Expressions

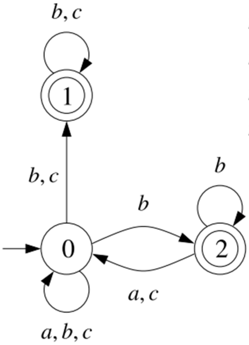
- Given a NBA A , we look at it as a NFA, and compute regular expressions $r_q^{q'}$.
- We show:

$$L_\omega(A) = L\Big(\sum_{q \in F} r_{q_0}^{q} \left(r_q^q\right)^\omega \Big)$$

  – An ω-word belongs to $L_\omega(A)$ iff it is accepted by a run that starts at q_0 and visits some accepting state q infinitely often.

From NBAs to ω-Regular Expressions

- Example:



$$r_0^1 = (a+b+c)^*(b+c)$$
$$r_0^2 = (a+b+c)^*b$$
$$r_1^1 = (b+c)^*$$
$$r_2^2 = b+(a+c)(a+b+c)^*b$$

$$L_\omega(A) = r_0^1\,(r_1^1)^\omega + r_0^2\,(r_2^2)^\omega$$