Lecture 1 (30-07-2021)

30 July 2021 09:28

Automata on infinite words

Some notations

- Let Σ be a finite nonempty set (called alphabet).
- A finite word (over Σ) is a finite (possibly empty) sequence of letters from Σ.

$$w = a_0 a_1 \cdots a_n, \qquad a_i \in \Sigma.$$

ϵ is the empty word.

Σ^* is the set of all finite words (over Σ).

- An infinite word over Σ is an infinite sequence of letters from Σ.

$$\alpha = a_0 a_1 a_2 \cdots, \qquad a_n \in \Sigma \quad \forall n \in \mathbb{N}_0.$$

(Different formality: $\mathbb{N}_0 = \{0, 1, \ldots\}$ and $\alpha : \mathbb{N}_0 \to \Sigma^*$)

Also, $\omega := \mathbb{N}_0$.

Σ^ω = all infinite words (on Σ)

(In general, given sets X and Y, Y^X denotes the set of all functions $X \to Y$)

Examples ① $\Sigma = \{a, b\}$

$\alpha = a\,b\,ab\,ab \cdots$

or: $\alpha(n) = \begin{cases} a & ; \ 2 \mid n \\ b & ; \ 2 \nmid n \end{cases}$

② $\alpha = abb\,abb\,abb$

$\alpha(n) = \begin{cases} a & ; \ 3 \mid n \\ b & ; \ 3 \nmid n \end{cases}$

$\alpha = (abb)^\omega$

③ $\gamma : \omega \to \{a, b\}$

$\gamma(n) = \begin{cases} a & ; \ n \text{ is prime} \\ b & ; \ \text{otherwise} \end{cases}$

$$\gamma = \underset{0}{b}\,\underset{1}{b}\,\underset{2}{a}\,\underset{3}{a}\,\underset{4}{b}\,\underset{5}{a}\,\underset{6}{b}\,\underset{7}{a}\,\underset{8}{b}\,\underset{9}{b}\,\underset{10}{b}\,\underset{11}{a} \cdots$$

γ has infinitely many 'a's and 'b's.
(Can't write as compactly as before.)

④ $\Sigma = \{a\}$, $\quad \Sigma^\omega = \{a^\omega\}$. $\quad (|\Sigma^\omega| = 1.)$

But if $|\Sigma| > 1$, then Σ^ω is <u>not</u> a countable set.

OTOH, Σ^* is always a countable set. $\quad (1 \leq |\Sigma| < \infty)$

<u>Automata</u>:

$$A = (Q, \Sigma, q_0, \Delta \subseteq Q \times \Sigma \times Q, \text{"Acceptance condition"})$$

→ a finite set of states

→ $q_0 \in Q$ → the initial state (unique)

→ $\Delta \subseteq Q \times \Sigma \times Q$ → the transition relation

$$(q, a, q') \in \Delta \equiv q \xrightarrow{a} q'$$

(Non-determinism here is fine)

Now, let $\alpha = a_0 a_1 a_2 \cdots \in \Sigma^\omega$ be given.

<u>A</u> run ρ of A on α is an infinite sequence of states

(run)

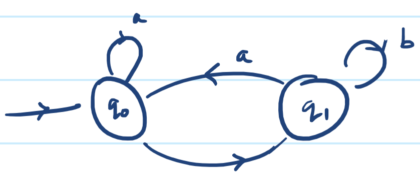
$$\rho = q_0 q_1 q_2 \cdots$$

Such that "q_0" is indeed the initial state and

$$\forall i \in \omega : \quad (q_i, a_i, q_{i+1}) \in \Delta$$

(In terms of functions: Given $\alpha : \omega \to \Sigma$, we have
$\rho : \omega \to \Sigma$ s.t. $\rho(0) = q_0$ and $(\rho(n), \alpha(n), \rho(n+1)) \in \Delta \ \forall n \in \omega$.)

Example.



$\alpha = (ab)^\omega$

$\rho = q_0 (q_0 q_1)^\omega$

$\rho = q_0 (q_0 q_1)^\omega$

"Acceptance condition": (Büchi automata)

$\alpha \to$ input word

$\rho \to$ a run of A on α

$\text{Inf}(\rho) :=$ the set of states which occur infinitely often along ρ

$= \{ q \in Q : \exists^\infty i \in \omega \text{ s.t. } \rho(i) = q \}$

Obs. $\text{Inf}(\rho) \neq \emptyset$. (There are only finitely many states.)

<u>Büchi automaton (BA):</u> Fix $G \subseteq Q$ called the "good states".

A run ρ is accepted by a BA if $\text{Inf}(\rho) \cap G \neq \emptyset$.

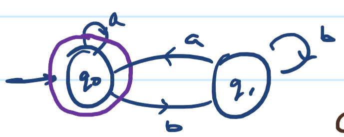
(Thus, some good state appears infinitely often.)

A word $\alpha \in \Sigma^\omega$ is accepted by A if α has an accepting run ρ on the word α.

$L(A) := \{ \alpha \in \Sigma^\omega : A \text{ accepts } \alpha \}$

$\hookrightarrow$ Language of A

Example

$A =$ 

$G = \{q_0\}$, $\Sigma = \{a, b\}$

Claim. $L(A) = \{ \alpha \in \Sigma^\omega : \alpha$ has inf.

many 'a' }.

Proof. Let the right side be L.

- $L(A) \subseteq L$:

  $\alpha \in L(A)$, $\alpha \equiv a_0 a_1 a_2 \cdots$

  Note that A is deterministic, thus α has a unique run ρ, which is accepted.

  $$\rho \equiv q_0 q_1' q_2' q_3' \cdots$$

  Thus, q_0 appear inf. often above. Since it only receives 'a', we see that 'a' appears inf. often.

- $L \subseteq L(A)$:

Let $\alpha \in L$. It has a unique run ρ.

Then, since α has inf. many 'a's, ρ will have inf. many 'q_0's. ☒

Can also write $L = (b^* a)^\omega$ once we have defined what that means.

Question: What about $\bar{L} = \Sigma^\omega \setminus L$? Can that be accepted by a Büchi automaton?

Lecture 2 (04-08-2021)

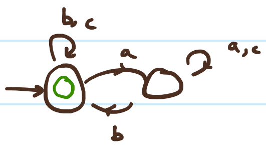
04 August 2021 09:31

Note. We do NOT allow ε transitions in this course.

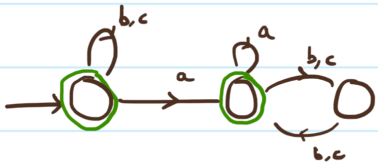
Fact. Even though we insisted on single initial state, the expressive power does not change if we allow more. (It is simply for convenience.)

Example (1) L_1 over $\Sigma = \{a, b, c\}$.

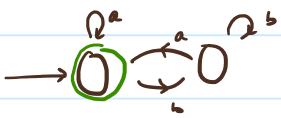
L_1 = every 'a' is eventually followed by a 'b'



(2) L_2 = any two occurences of 'a' are separated by even no. of other (b, c) letters

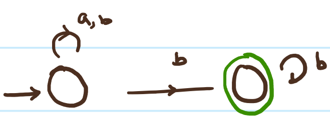


(3) $\Sigma = \{a, b\}$, L = inf. many 'a's



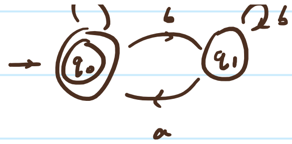
Complement : $\bar{L} = \Sigma^\omega \setminus L$ = finitely many 'a's

Q. What is a BA for $\bar{L}$?



Q. Do we have a deterministic Büchi automaton (DBA) for $\bar{L}$?

Note that we had an DBA for L :

Note that we had an DBA for L : 

A : $G = \{q_0\}$

Toggle states : A' : $G = \{q_1\}$

But $L(A') \supsetneq \bar{L}$.

$L(A')$ ↓ infinitely many b

$\bar{L}$ ↓ eventually b

$(ab)^\omega \in L(A')$ but $(ab)^\omega \notin \bar{L}$

Complementing the good state of a DBA A does <u>NOT</u> complement the accepted language.

<u>Claim.</u> There is No DBA A for $\bar{L} = \{\alpha \in \Sigma^\omega : \alpha$ has finitely many 'a'$\}$.

Thus, as opposed to finite languages, non-determinism actually gives us more languages.

<u>Proof.</u> We prove this by contradiction.

For the sake of contradiction, assume that $\exists$ DBA A such that $L(A) = \bar{L}$.

Suppose A has m states.

$\alpha_0 = b^\omega = b\,bb\cdots \in \bar{L}$

$\rho_0 = q_0\, q_1 q_2 \cdots$

↳ unique run of α_0

Since ρ_0 is accepting, $\exists\, n_1$ s.t. $q_{n_1+1} \in G$.

Pick the smallest such n_1.

$$\rho_0 = \overbrace{b\ b\ \cdots\ b}^{n_1} \mid \boxed{b}\ b\ \cdots$$

↑ first good state

Define $a_1 := b^{n_1}\, a\, b^\omega \in \bar{L}$.

$$\alpha_1 = b \cdots b^{n_1} \; a \; b \; b \cdots$$

$$\rho_1 = \square \cdots \square \; \square \cdots \square$$

$\in G$ again a good state

Then, we can get n_2 s.t. $b^{n_1} a b^{n_2} a$ ends at a good state.

Then, $\alpha_2 = b^{n_1} a b^{n_2} a b^{\omega} \in L$.

Its unique run ρ_2 matches ρ_1 until $b^{n_1} a b^{n_2} a$.

Keep getting $n_1, n_2, n_3, \ldots, n_{m+1}$.

$$\alpha_m = b^{n_1} a b^{n_2} a \cdots b^{n_{m+1}} a b^{\omega} \in L.$$

$$\rho_m = \underbrace{\square_{\in G} \; \square_{\in G} \; \cdots \; \square_{\in G}}_{m+1 \text{ states}}$$

By PHP, two of these $m+1$ good states are equal. Loop between them to get a word which is accepted but has inf. many 'a's. ∎

Cor. DBA $\subsetneq$ NBA in terms of expressiveness.

Def^n A language $L \subseteq \Sigma^{\omega}$ is said to be *ω-regular* if there exists a (possibly non-deterministic) Büchi automaton A such that $L(A) = L$.

Closure Properties of ω-Regular Languages

1) closure under union:

$L_1 = L(A_1)$, $\quad A_1 = (Q_1, q_0^1, \Sigma, \Delta_1, G_1)$,

$L_2 = L(A_2)$, $\quad A_2 = (Q_2, q_0^2, \Sigma, \Delta_2, G_2)$.

To-do: Construct a BA A s.t. $L(A) = L_1 \cup L_2$.

We do the usual product construction.

$$\left(Q_1 \times Q_2,\ (q_0^1, q_0^2),\ \Sigma,\ \Delta,\ G_1 \times Q_2 \cup Q_1 \times G_2\right)$$

$$(q_1, q_2) \xrightarrow{a} (q_1', q_2')$$

iff $\quad q_1 \xrightarrow{a} q_1'$ and $q_2 \xrightarrow{a} q_2'$.

$\alpha = a_0\ a_1\ a_2 \cdots$

$\rho^1 = q_0^1\ q_1^1\ q_2^1 \cdots$ a run of A_1 on α

$\rho^2 = q_0^2\ q_1^2\ q_2^2 \cdots$ — " — A_2 — " —

"$\rho^1 \times \rho^2$" $= \begin{pmatrix} q_0^1 \\ q_0^2 \end{pmatrix} \begin{pmatrix} q_1^1 \\ q_1^2 \end{pmatrix} \begin{pmatrix} q_2^1 \\ q_2^2 \end{pmatrix} \cdots$ a "product run" on α

Here we assume that each $\alpha \in \Sigma^\omega$ has at least one run on both A_i. (Can always ensure this by adding a dead state)

With the above assumption,

$$G = (G_1 \times Q_2) \cup (Q_1 \times G_2) \text{ gives}$$

the language as $L_1 \cup L_2$.

A simpler construction: